D.B.Baarda / A.van Londen

Leren omgaan met de ATARI 600XL/800XL

BASIC voor jonge computergebruikers

Kluwer Technische Boeken

Leren omgaan
met de
ATARI 600 XL/800XL

Leren omgaan met de ATARI 600 XL/800 XL

BASIC voor jonge computergebruikers

Drs. D. B. Baarda
en
Drs. A. van Londen

Kluwer Technische Boeken B.V. - Deventer - Antwerpen

Omslag en illustraties: W. Niessink

ISBN 90 201 1800 5
D/1985/0108/196

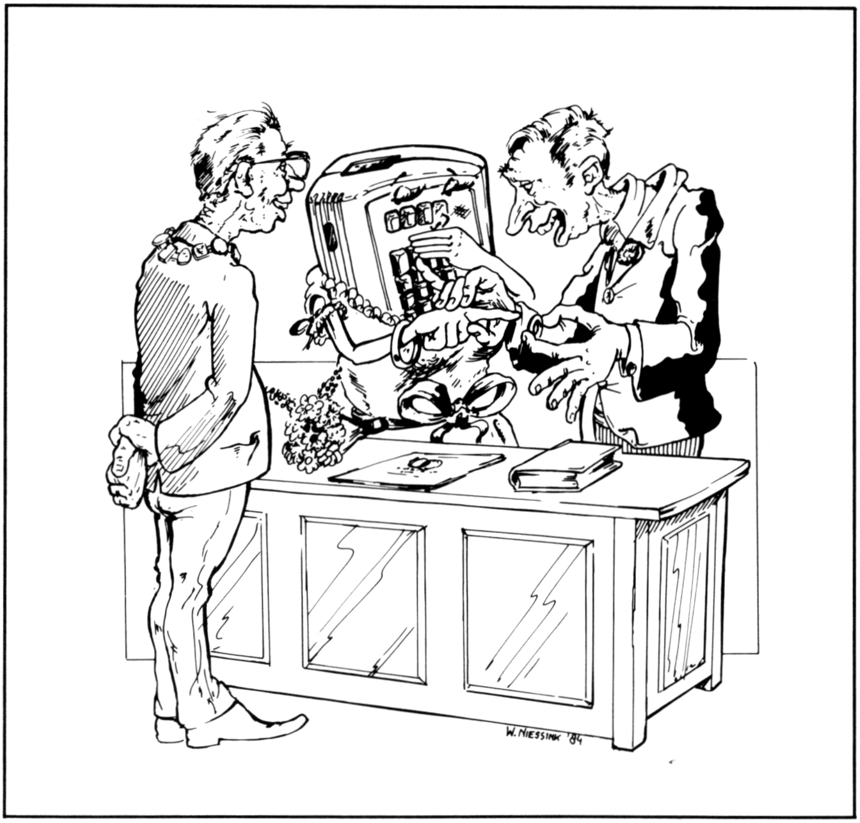
© 1985 Kluwer Technische Boeken B.V. – Deventer

1e druk 1985

Niets uit deze uitgave mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm of op welke andere wijze ook, zonder voorafgaande schriftelijke toestemming van de uitgever.

No part of this book may be reproduced in any form, by print, photoprint, microfilm or any other means without written permission from the publisher.

Ondanks alle aan de samenstelling van de tekst bestede zorg, kan noch de redactie noch de uitgever aansprakelijkheid aanvaarden voor eventuele schade, die zou kunnen voortvloeien uit enige fout, die in deze uitgave zou kunnen voorkomen.



Met dank aan:
Arleen
André
Christian
Jonathan
Mariska
Oscar
Raf
Ruth

De programma's zijn geschikt voor de meeste BASIC-versies en getest op de ATARI 600 XL

INHOUD

Waarover gaat dit boek?

Deel 1
WAT KAN EEN COMPUTER?
Tekenen met de computer 13
Schrijven met de computer.... 17
Rekenen met de computer 19
Nog meer tekenmogelijkheden.. 25
Kleuren en geluid maken met de computer 32
Kleuren met de computer 32
Geluid 34

Deel 2
PROGRAMMEREN IN BASIC
Klaar om te programmeren? ... 39
Regelnummer 39
RETURN 40
END 41
Opdrachten 41
NEW 41
LIST 42
RUN 43
BREAK 43
RESET 44

LEREN PROGRAMMEREN
Inleiding 47
PRINT 49
DIM 55
INPUT 59
GOTO 63
GOSUB en RETURN 67
REM 73
IF ... THEN 76
FOR ... NEXT 80
READ en DATA 85
RND 91
Fouten in je programma's 95

Deel 3
PROGRAMMA's
Rekenen met de computer 101
Toetsenbord 106
Demonstratie graphics 107
Kleurendriehoek 107
Random tekening 107
Old Mac ATARI 108
De computer is verliefd 108
Grootste Gemene Deler 109
Uitrollende dobbelsteen 109
Twee dobbelstenen 109
Getallen omkeren 110
Namen omkeren 110
Gedachtenlezen 111
Priemgetallen 112
Nonsens Generator 113
Telefoonboek 113
Rekenkundige tafels 114
Overhoren tafels 114

Bijlage 1
DE CASSETTERECORDER
Bewaren 117
Laden 118

Bijlage 2
ANTWOORDEN OP DE VRAGEN 120

WAAROVER GAAT DIT BOEK?

Het boek dat je nu begint te lezen gaat over het omgaan met computers, en het is speciaal voor kinderen geschreven. (Dit betekent niet, dat een volwassene er niets uit kan leren!) We leren je om zelf de ATARI te gebruiken. Als het goed gaat heb je er geen volwassene bij nodig.
In dit boek kun je BASIC leren. Dit is een computertaal. Daarover gaat het tweede deel van dit boek. Als je al kunt omgaan met het toetsenbord, kun je beginnen met het hoofdstuk KLAAR OM TE PROGRAMMEREN? Anders kun je beter vooraan beginnen en in ieder geval TEKENEN en SCHRIJVEN MET DE COMPUTER lezen voordat je aan het tweede deel begint.
Het eerste deel van het boek behandelt hoe je kunt tekenen en schrijven, kleuren en geluid maken met de computer. Zo gebruik je de computer als een soort schrijf- of tekenpapier, als kleurdoos of muziekinstrument. We leren je welke toetsen je hiervoor moet indrukken. Het verschil met papier is alleen, dat je tekening of verhaal weg is, als je de computer uitzet, en dat is soms jammer.
In het tweede deel kun je leren, hoe je de computer ook voor andere zaken kunt gebruiken. Daarbij kun je denken aan het onthouden en bewaren van dingen (bijvoorbeeld het opstel dat je wilt gaan schrijven of de tekening die je gaat maken, of het muziekstukje).
En hij kan ook snel karweitjes voor je uitvoeren waar je zelf veel meer tijd voor nodig hebt, zoals een moeilijke som uitrekenen, duizend strafregels schrijven, je huiswerk-hulp zijn, en spelletjes ontwerpen. Maar dit kan de computer niet zonder jouw hulp. Jij moet hem vertellen, wat je wilt dat hij doet. Je schrijft de computer daarvoor een soort 'brief' via het toetsenbord. Zo'n 'brief' noem je een 'programma'. Je kunt niet beginnen

met 'Beste computer, luister eens...', want dat snapt hij niet. Je moet het programma in zijn eigen taal schrijven, een computertaal. De computertaal die wij je leren heet 'BASIC'. BASIC is een schrijftaal. Er is niemand die BASIC spreekt. Er worden alleen computerprogramma's in geschreven. Dit leer je in het tweede deel: PROGRAMMEREN IN BASIC.

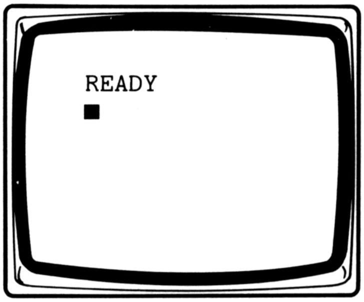
In het derde deel vind je voorbeelden van programma's die al gemaakt zijn. Je kunt ze overtypen. Als je daarna RUN typt en vervolgens op [RETURN] drukt, doet de computer wat er in het programma staat. Sommige van deze programma's worden in deel 1 beschreven.

DEEL 1

WAT KAN EEN COMPUTER?

TEKENEN MET DE COMPUTER

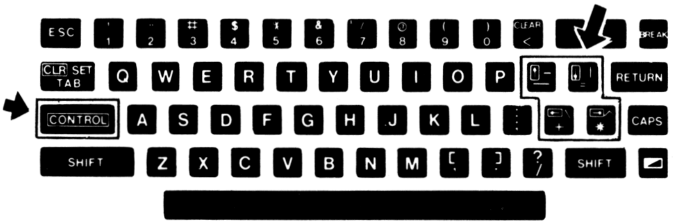
Als je de computer aanzet zie je als het goed is:



\# Je ziet niet zoals bij een gewoon TV-toestel een testbeeld. Wat je ziet is een TV-scherm waar de computer op kan tekenen of schrijven en waar jij op kunt tekenen of schrijven via het toetsenbord. Het lijkt dus gewoon op een blanco velletje papier. Er staat wel een blokje op. Daarmee geeft de computer aan, dat jij aan de beurt bent om iets op het scherm te zetten. Als de computer zelf iets gaat schrijven, verdwijnt dat blokje (de cursor heet die in het Engels). In dit deel van het boek zullen we vooral kijken wat jij op het scherm kunt tekenen en schrijven. Hoe je ervoor zorgt dat de computer iets gaat tekenen of schrijven leer je in het tweede deel. Maar het blokje staat er, dus jij kunt iets typen op de plek waar het blokje staat.
Druk maar eens op *.
Heb je het gedaan? Dan staat er als het goed is een sterretje op het TV-scherm.
Druk nog eens op *. Je ziet dat er nu nog een ster naast staat. Het blokje schuift iedere keer een plekje naar opzij als je iets intypt.
Het blokje is een soort stuur. Druk eens 35 keer op *. Als het goed is staat het blokje nu aan de rand en heb je een hele rij sterretjes. Klopt het? Nee, zet dan de computer uit en weer aan en begin nog eens bij #.

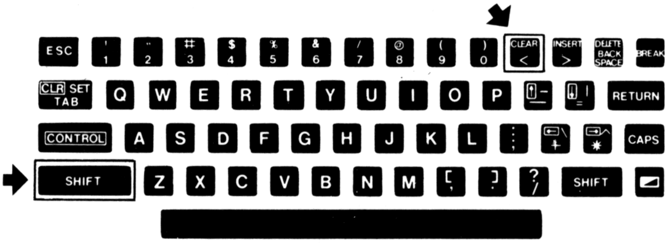
Heb je een hele rij sterretjes en staat het blokje aan de rand, typ dan nog eens *. Het blokje staat nu weer helemaal aan de linkerkant, onder wat je net getypt hebt. Als een regel vol is, gaat hij namelijk naar het begin van de volgende regel.

\# \# Je kunt het blokje ook besturen met de toetsen [CONTROL] en [↑] [↓] [←] en [→]. Kun je ze vinden?
[CONTROL] zit links en de pijltjes zitten rechts.



Druk eerst op [CONTROL]. Je moet je vinger op deze knop houden. Druk nu op [→]. Je ziet dat het blokje opzij schuift. Breng het maar eens naar de zijkant van het scherm.
Druk nu eens op [CONTROL] en tegelijk op [↓]. Nu gaat het blokje naar beneden. Probeer eens in de onderste hoek te komen.
Nu moet je weer omhoog. Daarvoor moet je weer twee toetsen tegelijk indrukken, namelijk [CONTROL] en [↑]. En wanneer je naar links wilt gebruik je [CONTROL] en [←].
Kijk nu eens of je in alle hoeken van het TV-scherm kunt komen. Lukt het niet, lees dan nog eens vanaf # #.
We gaan nu tekenen, waarbij het blokje het stuur voor het potlood is.

\# \# \# Als je eerst op [SHIFT] en tegelijk op [CLEAR] drukt krijg je een heel schoon scherm.
Kun je de toetsen vinden?

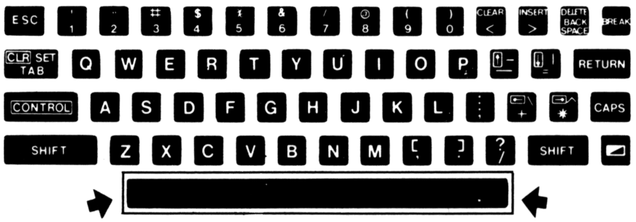


Onthoud het maar goed.
Dus wanneer je op [SHIFT] en [CLEAR] drukt, is het TV-scherm schoon.
Wij beginnen met een vierkant in de hoek van het TV-scherm. Typ eerst vijf keer *. Nu moet je proberen onder het laatste sterretje te komen.
Dus eerst [CONTROL] en [↓]. Je gaat nu één regel naar beneden. Dan ga je met [CONTROL] en [←] een plekje terug.
Als het goed is sta je nu onder het laatste sterretje.
Staat het blokje onder het laatste sterretje?
Nee, begin dan nog eens bij # # #.
Probeer nu dit vierkant te maken.

```
* * * * *
*       *
*       *
*       *
* * * * *
```

UITVEGEN

Het is erg moeilijk en je hebt misschien wel een fout gemaakt die je wilt uitvegen. Ga dan met het blokje naar wat je wilt uitvegen en druk op die hele lange toets onderaan. De spatiebalk noemt men die.



Veeg nu eens de onderste lijn van het vierkant uit. Gelukt? Nee, lees dan nog eens # # # #.
Je kunt nu van alles tekenen. Wij hebben een aantal voorbeeldjes gemaakt. Maar je kunt ook allerlei dingen zelf bedenken. Veeg eerst het TV-scherm maar schoon. Weet je niet meer hoe dat moet, kijk dan bij # # #.

opgaven

1 Maak eerst het scherm schoon. En teken dan van sterretjes het volgende huisje.

```
    *
  *   *
*       *
*       *
*       *
* * * * *
```

2 Maak het scherm weer schoon. En teken dan weer van sterretjes het beest dat je hieronder ziet staan. Het is een soort computerhond.

```
          *
          * *
*         * * *
  * * * * *
  * * * * *
  * * * * *
  *       *
  *       *
```

3 Maak het scherm schoon. En teken een boom, boot, auto of iets wat je zelf bedenkt.

SCHRIJVEN MET DE COMPUTER

Het zal je niet verbazen dat je ook kunt schrijven met de computer. Hij lijkt immers erg veel op een typemachine. Je kunt hem dan ook gebruiken om te typen. Op de ATARI zitten alle lettertoetsen op dezelfde plaats als bij een gewone typemachine. Je kunt er bijvoorbeeld je naam op typen.

\# Typ maar eens OSCAR of jouw naam.

Je ziet, het zijn alleen maar hoofdletters. De computer kent ook wel kleine letters.

Druk maar eens op [CAPS] en typ dan oscar. Als het goed is, dan staat er nu oscar. Als je [CAPS] hebt ingedrukt kun je kleine letters typen. Je zult zeggen ik heb hoofdletters en kleine letters door elkaar nodig. En dat kan ook. Als je [SHIFT] en bijvoorbeeld [A] tegelijk indrukt krijg je de hoofdletter A. Druk je alleen op [A] dan krijg je a op het TV-scherm.

Druk eens 5 keer op [A]. Krijg je nu aaaaa? Als je dit niet hebt maar wel AAAAA, druk dan nog eens op [CAPS] en druk 5 keer op A. Nu moet het kloppen.

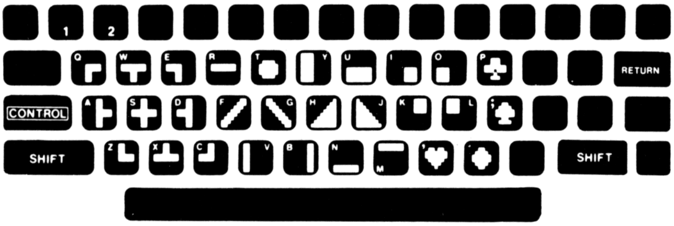
Druk eens 5 keer op [SHIFT] en [A] tegelijk. Nu krijg je AAAAA. Lukt het niet ga dan nog eens terug naar #.

Vooral op de cijfertoetsen zie je telkens boven het cijfer nog iets anders staan. Boven de 2 staat bijvoorbeeld " en boven 9 staat (. In programma's die je gaat maken kunnen deze tekens erg belangrijk zijn. Zoek ze maar eens op. Je kunt " krijgen door [SHIFT] en [2] tegelijk in te drukken.

Ook het vraagteken kun je alleen door [SHIFT] en [/] samen in te drukken krijgen. Als je [SHIFT] vergeet krijg je /.

Onderaan het toetsenbord vind je een hele lange toets, dit is de spatiebalk. Daar kun je spaties mee maken, druk er maar eens op. Zoals je ziet schuift de cursor (het blokje □) een plaats op, hij typt niets. Er wordt een soort

van opening gemaakt, men noemt dit een spatie. Je vindt die bijvoorbeeld tussen twee woorden in.



Fouten verbeteren kun je met de cursorbesturing zoals wij hiervoor hebben uitgelegd. Het kan echter ook met de DELETE BACK SPACE . Druk er maar eens op. De cursor gaat nu een plekje terug. Typ eens FIETZ ; ja met een Z . Zoals je weet moet het FIETS zijn. Ga nu eens met DELETE BACK SPACE terug en maak van de Z een S . Je hoeft alleen maar op de S te drukken, de Z veegt hij vanzelf uit.
Het valt waarschijnlijk niet mee, om snel de verschillende letters te vinden. Wij hebben een programmaatje gemaakt dat je daarbij kan helpen. Het staat achterin en het heet 'TOETSENBORD'. Misschien kan iemand anders het even voor je intypen. Wij hebben het met opzet kort gehouden. Het programma geeft telkens een letter op en jij moet die dan zo snel mogelijk intypen. Aan het eind zie je hoeveel je er goed had.

opgaven

1 Typ eens: Ik leer al aardig typen. Het gaat goed!

Je moet zowel de hoofdletters als kleine letters en ook het uitroepteken (!) typen.

2 Typ eens:

2 " 5 $ ( ) ?

3 Oefen met het programma 'TOETSENBORD'.

4 Maak eens een gedicht op de computer.

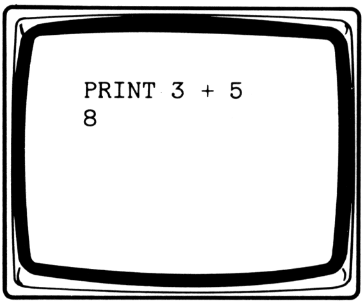
REKENEN MET DE COMPUTER

Computers werden vroeger alleen maar als rekenmachines gebruikt: rekenen kunnen ze veel sneller en nauwkeuriger dan mensen! Je zult zien hoe snel de computer je rekenhuiswerk kan maken. Je hoeft alleen maar je som in te typen, en hij weet het antwoord al. Als je voor je som PRINT typt, dan schrijft hij het antwoord op het TV-scherm.
De computer gebruikt voor vermenigvuldigen en delen niet dezelfde tekens die jij op school leert. Hieronder komt een lijstje:

	BASIC wat de computer begrijpt	*Op school leer je*
Optellen	+	+
Aftrekken	–	–
Vermenigvuldigen	*	×
Delen	/	:

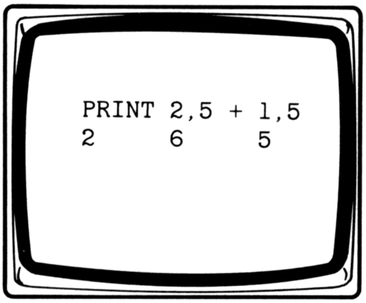
Als je de computer als rekenmachine wilt gebruiken, dan typ je PRINT en daarachter de getallen en het teken voor optellen, aftrekken of iets anders. Laten we een paar voorbeelden maken.

typ



– *druk op* RETURN

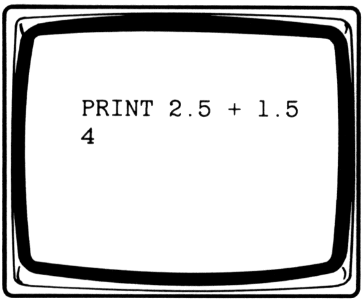
typ



– druk op RETURN

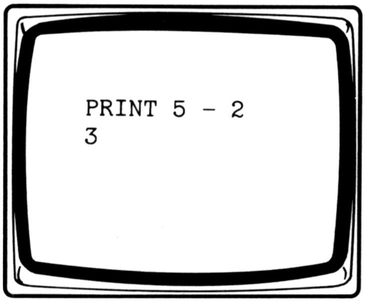
Hé, dat gaat fout. Je dacht dat er 4 zou komen en er staat nu 2 6 en 5. Hoe zou dat komen? De computer gebruikt de , (komma) voor iets anders dan waar wij hem nu voor nodig hebben. De komma betekent dat de volgende uitkomst een stukje verder op het scherm komt. Eerst de 2, dan een stukje verder 5 + 1 en weer een stukje verder 5. We hebben nu een . (punt) nodig. Dus onthouden: EEN . (PUNT) IN PLAATS VAN EEN , (KOMMA) midden in een getal.

typ



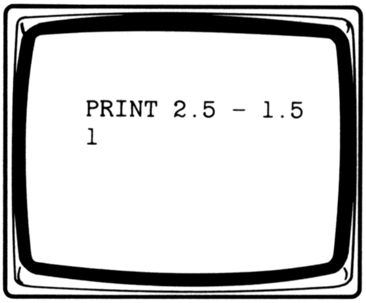
– Let op een punt in plaats van een komma

typ



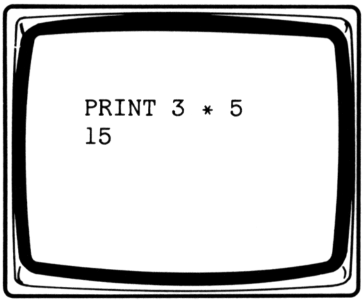
– Aftrekken

typ



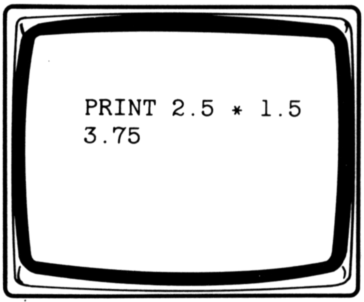
– Let op een punt in plaats van een komma

typ

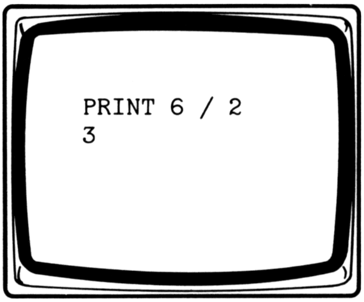


*– Vermenigvuldigen. Let op een * i.p.v. een ×*

typ

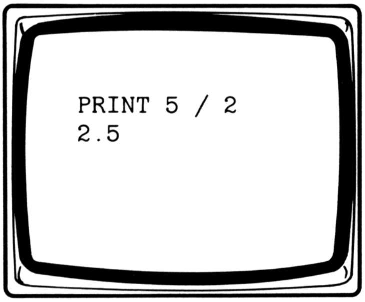


typ



– Delen. Let op een / i.p.v. :

typ

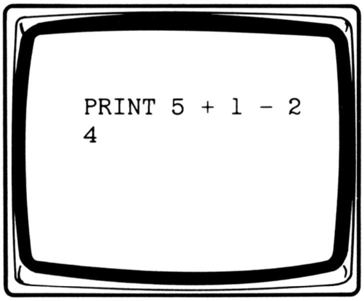


Als je deze voorbeelden gemaakt hebt, dan weet je hoe je de computer als rekenmachine kunt gebruiken.

Laten wij eens een som gaan maken:

Je hebt 5 gulden. Je krijgt er één gulden bij. Je koopt een boekje van 2 gulden. Hoeveel heb je over?
De computer kan hier snel bij helpen:

typ



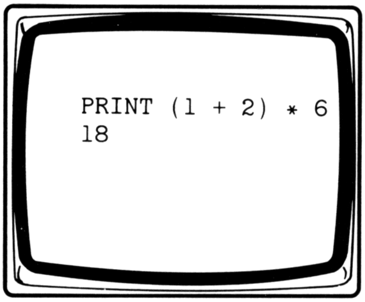
Dus 4 gulden over. Heb je in de gaten, dat je 5 + 1 − 2 gewoon achter elkaar kunt zetten?

Als je langere sommen maakt, moet je net als op school, haakjes gebruiken.

Voorbeeld: Jan heeft 6 gulden en Kees en Ans hebben ieder 6 gulden. Samen hebben ze:

(1 + 2) * 6 = 18 gulden   (*goed*)
1 + 2 * 6 = 13 gulden   (*fout*)

typ



De mogelijkheden voor rekenen, boekhouden, wiskunde, enz. van de computer zijn onbegrensd. Maar dat is te veel om uit te leggen.

De volgende fouten komen vaak voor als je gaat rekenen:

---

1 Een komma in plaats van een punt getypt.

`PRINT 2,5 + 1,5` *(fout)*

Als je dit typt krijg je drie getallen (2 6 5).
Je hebt in plaats van een punt een komma gezet.

`PRINT 2.5 + 1.5` *(goed)*

2 Aanhalingstekens gebruikt.

```
PRINT "5 + 3 "
5 + 3
```
(fout)

Wat tussen aanhalingstekens (") wordt gezet komt precies zo op het TV-scherm. En dat was niet de bedoeling. Je wou 5 + 3 laten uitrekenen. Dus:

```
PRINT 5 + 3
8
```
(goed)

---

opgaven

De oplossingen vind je achterin.

1 Hoeveel is

7	+	6
18	+	33
9	−	3
19	−	17
2	*	4
12	*	15
4	/	2
32	/	8

2 Tel 7 en 6 bij elkaar op en trek er 3 vanaf.

3 Je hebt f 2,50 zakgeld. Je geeft drie kwartjes aan je vriend. Hoeveel heb je over?

4 Drie zusjes willen samen een computerspel kopen van f 50,00. Ze krijgen ieder f 12,50 van een tante. Daarvan moet van hun ouders f 3,00 naar de bank. Hoeveel moeten ze nog sparen?

NOG MEER TEKENMOGELIJKHEDEN

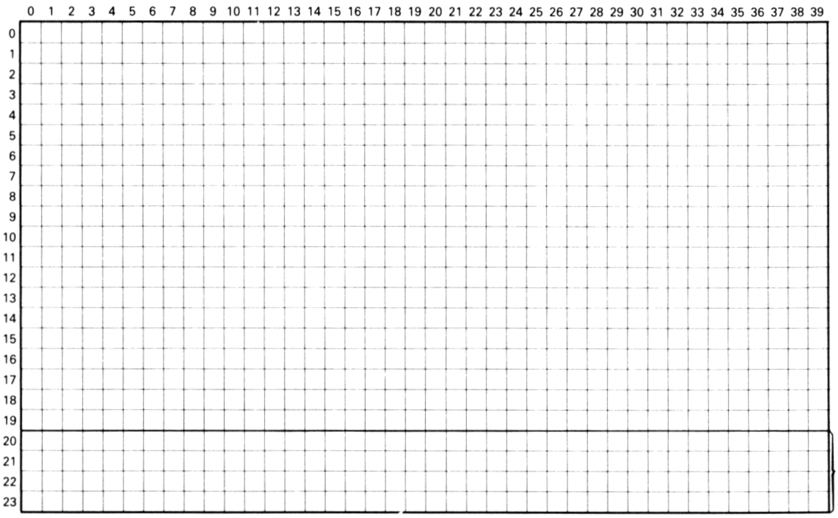
Misschien ben je al eens een grappig figuurtje tegengekomen. Dat zijn de grafische tekens. Je kunt ze krijgen door op [CONTROL] te drukken.
Houd [CONTROL] eens vast en druk op Q, W, E, R en T

je ziet nu

Er zitten nog meer van deze grappige figuurtjes op je toetsenbord en je kunt er mooi mee tekenen. Je krijgt ze door ze samen met de [CONTROL] -toets in te drukken.

Op deze tekening zie je waar ze zitten:

Zoek de eens op en de .
Heb je ze gevonden?
Je kunt hier prachtig mee tekenen.
Probeer maar eens een hekje, boompje of huisje te maken.
Je kunt met je computer ook lijnen of figuren tekenen net als met een potlood. Alleen heb je daar een ander scherm voor nodig. De computer gebruikt net als jij om te schrijven iets anders dan om te tekenen.
Een van de tekenschermen kun je krijgen door GR.3 te typen.
Eigenlijk moet je GRAPHICS 3 typen maar afkorten is veel makkelijker. Scherm 3 ziet er zo uit:



Je kunt met dit scherm zowel schrijven als tekenen. In de onderste 4 regels kun je schrijven. Wij hebben er een hok omheen gezet. In het bovenste deel kun je tekenen.

\# Heb je al GR.3 getypt? Zoniet dan nu even doen.
Zo nu gaan we eerst een goud/geel potlood nemen.

typ `COLOR 1`

Je moet nu eerst zeggen waar je wilt beginnen. Dat doe je met PLOT.

typ `PLOT Ø,Ø`

Zo, nu zie je links helemaal bovenin een blokje. Bij sommige televisies valt het net buiten het scherm, probeer dan PLOT 1,0.
Je kunt ook zo'n blokje helemaal rechts bovenin zetten

typ `PLOT 39,Ø`

Zo nu staat er helemaal rechts bovenin een blokje.
Je moet altijd eerst één van de cijfers van de bovenkant nemen en dan één van de zijkant.
Weet je nu waar je terecht komt als je PLOT 39,19 typt? Probeer maar.

typ `PLOT 39,19`

Juist dat is het meest rechtse blokje boven het schrijfdeel, daar staat nu ook een goud/geel blokje. Als je een goud/geel blokje links onderaan wilt zetten hoe doe je dat dan?
Probeer maar.

typ `PLOT Ø,19`

Lukt het nog niet dan kun je op [RESET] drukken en nog eens beginnen bij #. [RESET] is één van de zilverkleurige toetsen.

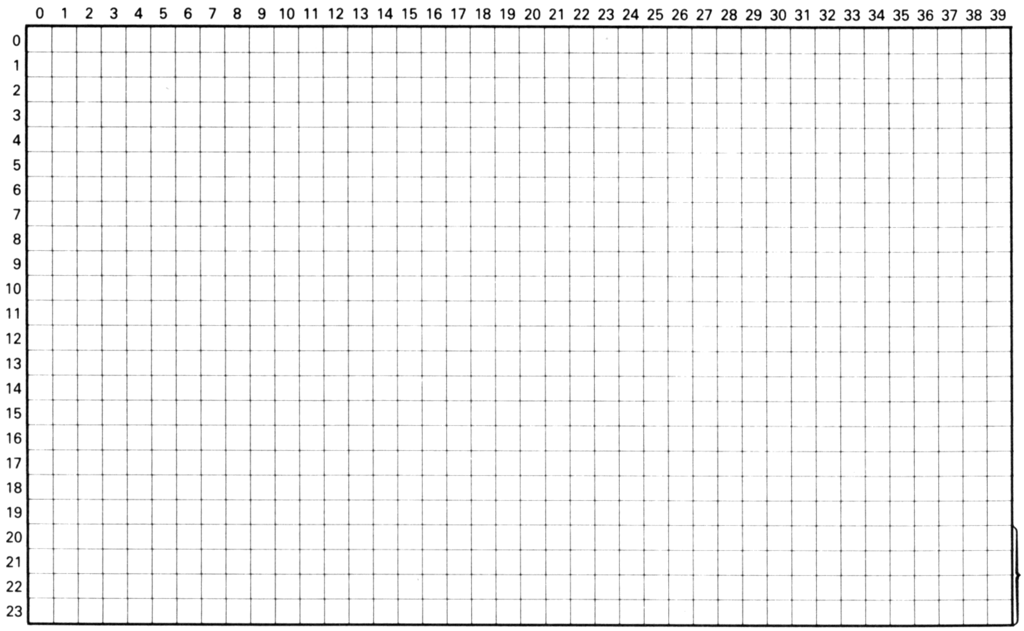
Laten we eens proberen een driehoek te tekenen. Druk eerst op [RESET].

typ	`GR.3`	
typ	`COLOR 1`	*is een goud/geel potlood*
typ	`PLOT Ø,Ø`	*begin links bovenin*
typ	`DRAWTO 39,Ø`	

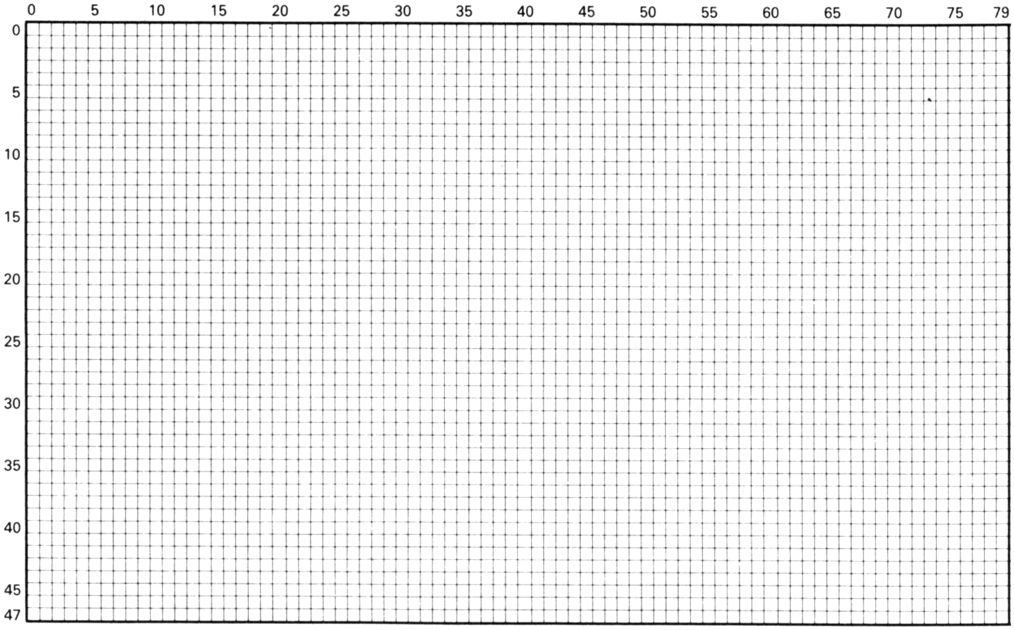
Hó even. Wat is dat nou DRAWTO? Dit zijn twee Engelse woordjes en ze betekenen TEKEN NAAR. Met DRAWTO kun je tegen de computer zeggen waar hij een lijn moet trekken. En wat 39,0 betekent weet je al, want dat is rechts boven. Dus eigenlijk staat er zoiets als GA NAAR. Ga met je potlood naar rechts boven. Probeer maar en je zult zien dat alle hokjes tussen 0 en 39 goud/geel gemaakt worden.

typ	`DRAWTO 39,19`	*ga naar rechts onder*
typ	`DRAWTO  Ø,Ø`	*ga naar links boven*
typ	`PRINT "dit is een gekke driehoek"`	

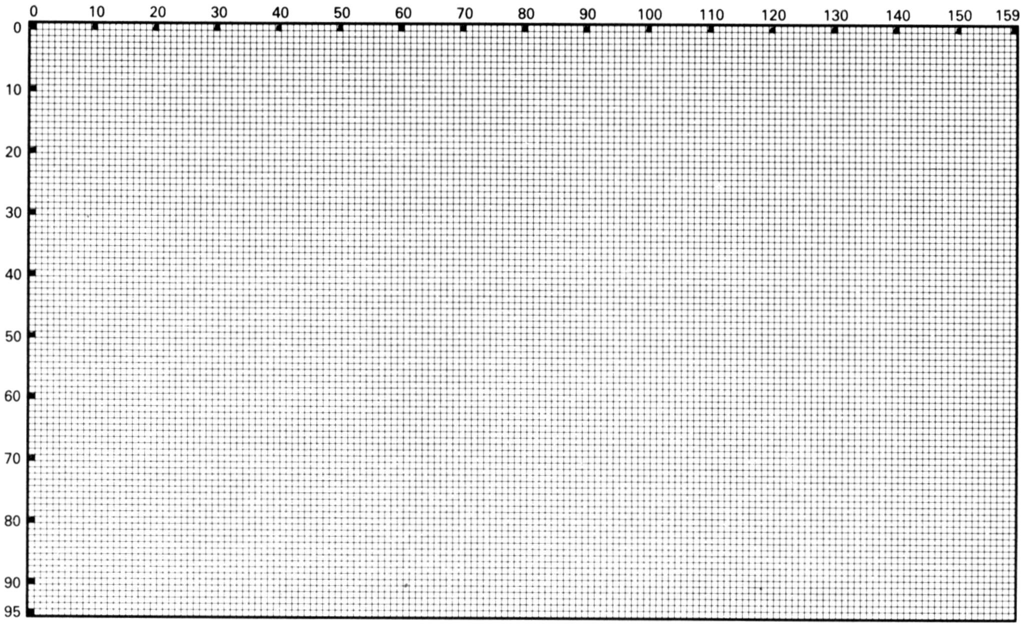
Zo nu weet je ook gelijk dat je kunt schrijven. Je kunt er tekst onder zetten.
Het is inderdaad een gekke driehoek, de schuine lijn is niet erg mooi. Dit komt omdat de hokjes die bij papier 3 gekleurd worden nogal groot zijn.
Bij andere GRAPHICS (papiersoorten) kun je meer en kleinere hokjes krijgen. Op de volgende pagina's zie je de voorbeelden van de andere GRAPHICS.



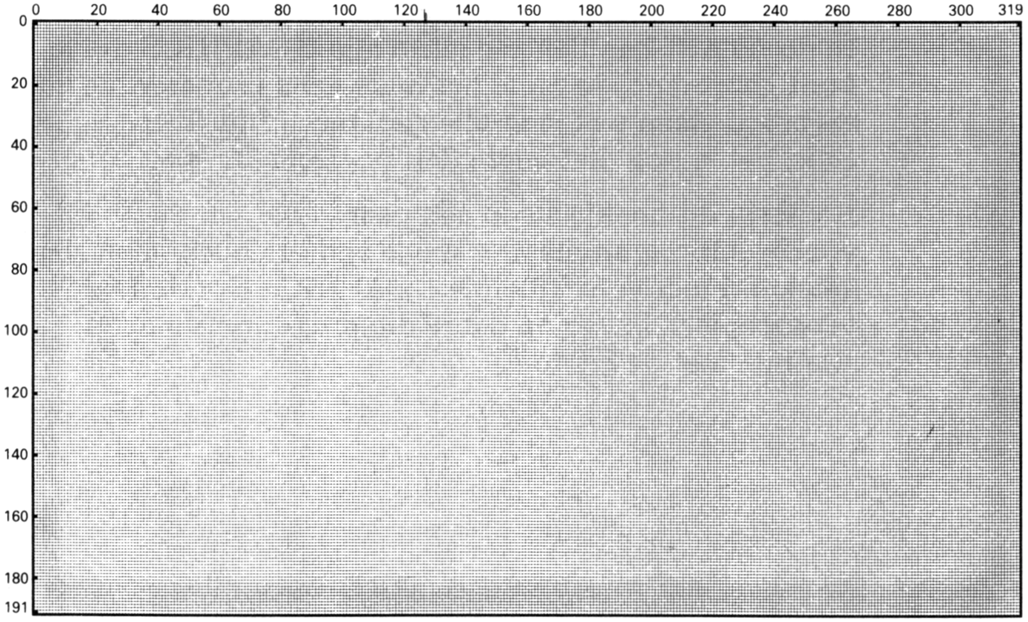
graphics 3



graphics 4



graphics 6



graphics 8

Je ziet dat GRAPHICS 8 zelfs 320 hokjes op de bovenste rij heeft. Bij GRAPHICS 3 waren dat er 40. We zullen de driehoek nu ook eens op papier 8 maken. De getallen bij PLOT en DRAWTO worden nu alleen anders. Het bovenste hokje rechts wordt nu 319,0 in plaats van 39,0. Het meest rechtse hokje boven het schrijfdeel wordt nu 319,159 in plaats van 39,19.

Typ de volgende regels maar in.

Druk eerst op

typ	`GR.8`	*papier 8*
typ	`COLOR 1`	*goud/geel potlood*
typ	`PLOT Ø,Ø`	*links bovenin beginnen*
typ	`DR. 319,Ø`	*DR. is afkorting van DRAWTO*
typ	`DR. 319,159`	*rechts onderin*
typ	`DR. Ø,Ø`	
typ	`PRINT "dit is een mooie rechthoek"`	

Dit is echt een mooie rechthoek. Door die kleine hokjes lijkt de schuine lijn een echte lijn.

In de ATARI-handleiding staan alle GRAPHICS die je kunt gebruiken. Je moet het maar eens proberen. Je zult verbaasd staan wat je er allemaal mee kunt doen.
Lukt het je nog niet begin dan nog eens bij ##.

Achterin dit boekje staan een paar voorbeeldprogrammaatjes die je een driehoek laten zien, getekend met verschillende GRAPHICS (soorten tekenpapier). Let er op dat je die programma's moet stoppen met de RESET-toets.

opgaven

1 Zoek de volgende tekens eens op en typ ze dan op het scherm , , , en .

2 Je kunt met deze tekens weer allerlei leuke figuren en patronen maken. Bijvoorbeeld een dambord met en Probeer het maar eens.

3 Wij hebben wat voorbeeldjes gemaakt van wat je zou kunnen maken. Teken ze eens na. En probeer er zelf ook een paar te bedenken.

KLEUREN EN GELUID MAKEN MET DE COMPUTER

De ATARI heeft twee mogelijkheden die je niet op alle huiscomputers vindt: aangesloten op een kleurentelevisie kan het scherm worden 'gekleurd' en je kunt de computer ook geluid laten maken. We zullen in de volgende hoofdstukken de belangrijkste dingen leren over het gebruik van kleuren en geluiden. Maar ook hiervoor geldt dat de mogelijkheden van de computer haast onbeperkt zijn, en dat je daar alleen maar achter kunt komen door zelf te experimenteren. Wij zullen je laten merken dat er veel kan.

KLEUREN MET DE COMPUTER

Zoals je misschien zelf al ontdekt hebt kun je verschillende kleuren krijgen, als je de computer op een kleurentelevisie hebt aangesloten.
Je kunt dat regelen met SETCOLOR of afgekort SE.
(vergeet de punt niet).
Typ bijvoorbeeld maar eens `SE. 2,12,6`
Vergeet niet op [RETURN] te drukken.
Als het goed is wordt je scherm nu groen.

Je ziet dat er achter SETCOLOR, 3 getallen staan. Het eerste getal geeft aan uit welke verfpot je de kleur haalt. Er zijn 5 verfpotten die genummerd zijn van 0 tot en met 4. Als je pot 4 gebruikt in plaats van 2 wordt ook de rand van het scherm groen.
Typ maar eens `SE. 4,12,6`
Niet alle potten zijn altijd bruikbaar. Op dit moment geven alleen de potten 2 en 4 een kleur. Pot 2 geeft het scherm een kleur en pot 4 geeft de rand een kleur. Door iets te veranderen kun je weer andere potten gebruiken maar daar komen wij straks wel op. Het tweede getal, de

12, bepaalt de kleur.
Groen is 12 en bijvoorbeeld paars is 4.
Typ maar eens `SE. 2,4,6`
Het scherm wordt nu paars.

Hieronder staan de kleuren met hun cijfers.

0 = grijs
1 = goud/geel
2 = oranje
3 = rood
4 = paars
5 t/m 9 zijn allerlei kleuren blauw
10 = groen/blauw
11 t/m 13 zijn allerlei soorten groen
14 = geel/groen
15 = oranje/geel

Je kunt al deze kleuren ook in licht en donker krijgen. Je regelt dat met het laatste getal. Als je 0 typt is het donker en als je 14 typt is het heel erg licht. Dus `SE. 2,12,0` is donker groen en `SE. 2,12,14` is heel licht groen. Je hoeft niet alleen 0 of 14 te gebruiken, je mag ook 2 (een beetje donker) 4 (iets minder donker) 6, 8, 10 of 12 gebruiken. Je hebt het waarschijnlijk zelf al ontdekt, je mag hier alleen even getallen gebruiken. Nu nog even alles op een rijtje.

`SETCOLOR 2,12,2`

nummer van de verfpot	*de kleur*	*licht-donker*
normaal 2 voor scherm en 4 voor rand	in dit geval 12 = groen	0 = heel erg donker 14 = heel erg licht

Wij hebben een programmaatje gemaakt dat je alle kleuren laat zien. Typ het maar eens in.

```
10 FOR KL = 0 TO 15
20 FOR LD = 0 TO 14 STEP 2
30 SE. 2,KL,LD
40 PRINT "KLEUR= ";KL;" LICHTHEID= ";LD
50 FOR P = 1 TO 400 : NEXT P
60 NEXT LD
70 NEXT KL
80 END
```

typ `RUN`

Je kunt nu hetzelfde doen voor de kleuren van de rand. Verander in regel 30 de 2 maar eens in een 4.

GELUID

Typ het volgende programma eens in

```
10 FOR T = 255 TO 1 STEP - 5
20 SOUND 3,T,10,10
30 FOR PAUZE = 1 TO 10 : NEXT PAUZE
40 NEXT T
50 END
```

Als je nu RUN typt, hoor je het geluid van een opstijgend ruimteschip. Dat geluid wordt gemaakt met

```
SOUND 3,T,10,10
```

Je ziet dat er achter SOUND 4 getallen staan. Het eerste getal, de 3, is voor de stem die je wilt gebruiken. De ATARI heeft 4 stemmen die je kunt gebruiken.
Het voordeel van meer stemmen is dat je bijvoorbeeld stem 1 tegelijk met stem 2 een geluid kunt laten maken. Waarbij stem 1 een ander geluid maakt dan stem 2. Probeer het volgende programma maar eens.

```
10 X = 10 : Y = 10
20 PRINT "WIL JE STEM 1, STEM 2 OF ALLEBEI"
30 PRINT "TYP EEN 1, 2 OF 3 VOOR ALLEBEI"
40 INPUT A
```

```
 5Ø IF A = 1 THEN Y = Ø
 6Ø IF A = 2 THEN X = Ø
 7Ø SOUND 1,251,1Ø,X
 8Ø SOUND 2, 47,1Ø,Y
 9Ø FOR PAUZE = 1 TO 1ØØØ : NEXT PAUZE
1ØØ END
```

Heb je de twee stemmen gehoord?
Je kunt dus maximaal 4 stemmen gebruiken.

stem 0
stem 1
stem 2
stem 3

Je geeft ze aan met het eerste getal achter SOUND.

Het tweede getal is de toonhoogte. Net als jijzelf kan de computer hoog of laag zingen. Hij zingt op zijn laagst als je 255 invult en het hoogste als je 0 invult.

255 (*laag*) --- (*hoog*) 0

In het ruimteschip-programmaatje ging de toon van laag naar hoog, hij ging immers van 255 naar 1 in stapjes van 5.
In het programma met de 2 stemmen, gaf stem 1 een laag geluid (251) en stem 2 een hoog geluid (47).
Met het volgende programmaatje kun je de toonhoogten van de computer horen van hoog naar laag. Hij geeft ook telkens het cijfer van de toonhoogte.

```
1Ø FOR T = Ø TO 255 : STEP 5
2Ø PRINT "TOON " ; T
3Ø SOUND 1,T,1Ø,1Ø
4Ø FOR PAUZE = 1 TO 2ØØ : NEXT PAUZE
5Ø NEXT T
6Ø END
```

Nu weet je wat het tweede getal betekent.

Het derde getal is voor de vervorming. Je eigen stem kan ook een beetje vervormen, bijvoorbeeld als je erg

verkouden bent wordt je stem een beetje anders. Alleen met het getal 10 of 14 krijg je mooie stemmen. Als je bijvoorbeeld 8 of 6 (altijd even getallen) invult hoor je gekraak. Dat is handig als je het geluid van een botsing na wilt maken. Verander het programma op pagina 35 maar eens door voor de eerste 10 een 8 in te vullen. Als je nu RUN typt hoor je wat wij bedoelen.

Nu het laatste getal 10 nog. Dat is het volume. Daarmee kan de computer hard of zacht gezet worden. Met 15 is hij op zijn hardst en met 0 op zijn zachtst.

0 (*zacht*) -- (*hard*) 15

Met 0 hoor je helemaal niets meer, hierdoor kun je hem zijn mond laten houden. Hij maakt namelijk geluid, totdat jij zegt dat hij moet stoppen. Dat doe je door het volume op 0 te zetten. Hij stopt ook als hij END tegenkomt.

Dus alles op een rijtje

```
SOUND 1, 251, 1Ø, 15
```

1	251	1Ø	15
stem mag 0-3 zijn	*toon mag zijn 0-255*	*mooie stem mag zijn 10 of 14*	*volume mag zijn 0-15*
	251 = laag 0 = hoog	krakend 2,4,6,8,12	hard = 15 zacht = 0

DEEL 2

PROGRAMMEREN IN BASIC

KLAAR OM TE PROGRAMMEREN?

Je weet nu hoe je via het toetsenbord kunt tekenen en schrijven. Wat je op het toetsenbord intypt, kun je op het TV-scherm krijgen. Ook weet je hoe je letters en getallen op het TV-scherm kunt veranderen of uitvegen. Dat zijn de belangrijkste dingen die je moet weten om de fantastische wereld van het programmeren binnen te stappen. Als je kunt programmeren, kun je de computer opdrachten gaan geven die hij voor jou moet gaan uitvoeren. Je kunt hem snel laten rekenen, je kunt hem dingen laten onthouden zodat hij je bijvoorbeeld kan overhoren, je kunt spelletjes ontwerpen en nog veel, veel meer.

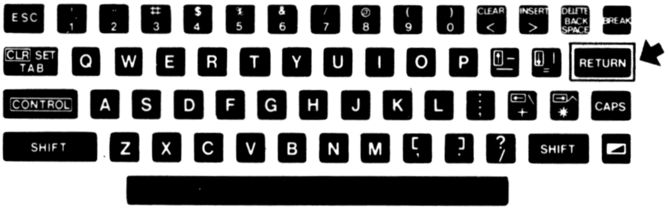
Wat is programmeren?
Programmeren is het schrijven van een programma, een soort brief, waarin precies staat wat de computer moet gaan doen. Zo'n programma bestaat uit regels, die allemaal moeten beginnen met een nummer.

REGEL-NUMMER

Zo'n nummer noemen we een REGELNUMMER. De regelnummers geven aan in welke volgorde een programma uitgevoerd moet worden. Je zou daarvoor nummer 1, 2, 3 enz. kunnen nemen. Als je de computer dan de opdracht geeft om een programma te gaan uitvoeren begint hij bij regel 1, dan regel 2, dan regel 3, enz.
Nu is programmeren een heel creatief en precies werkje. Vaak wil je tijdens het werken aan je programma tussen twee regels nog een regel zetten.
Maar wanneer je als eerste regel 1 hebt gekozen en als tweede regelnummer 2, kun je er niets meer tussen zetten. Je mag alleen hele nummers gebruiken. Daarom beginnen programma's bijna allemaal met regelnummer 10 en daarna 20, 30, 40, enz. Later kun je dan gemakkelijk tussen de eerste regel 10 en de tweede regel 20 nog een opdracht zetten op bijvoorbeeld regel 12, 15

of 18. Net zoals je een regel moet beginnen – met een heel nummer – moet je ook iedere regel eindigen.

RETURN Je beëindigt een regel altijd, door op de [RETURN] te drukken. Dat is de toets die rechts boven zit.



De [RETURN] toets is heel belangrijk. Alles wat je op het TV-scherm typt krijgt pas betekenis voor de computer als je op [RETURN] drukt. Dan pas gaat de computer controleren of wat op die regel staat ook zo in zijn geheugen mag. Na het indrukken van [RETURN] gaat de computer heel snel kijken of er een regelnummer voorstaat, of er begrijpelijke opdrachten tussen staan en nog veel meer. Als alles goed gaat, accepteert de computer die regel en stopt hem in zijn geheugen. Op het scherm is die controle niet te zien. Je ziet alleen maar dat het blokje (de cursor) weer helemaal links staat. Je kunt dan een nieuwe regel intypen. Als het niet goed gaat schrijft de computer achter het regelnummer een ERROR. Dat betekent dat je iets fout hebt gedaan. Je kunt de regel het beste nog eens intypen.
De computer kan op een regel vrij veel tekens onthouden. Meestal stoppen wij er maar 10 of 20 tekens in. Een aantal van die regels vormen dan samen een programma. Je moet dus onthouden dat een programma in BASIC er altijd zo uit moet zien:

---

regelnummer
bijv. 1Ø . . .   *iedere regel eindigen*
       2Ø . . .   *met een druk op* [RETURN]
       3Ø . . .
      1ØØ END

END Zoals je ziet eindigen we op de laatste regel (in dit geval 100) met END. END is het Engelse woordje voor einde. Als de computer END tegenkomt, weet hij dat het programma klaar is. Soms staat END niet aan het eind van een programma maar dat is eigenlijk niet mooi. Mooie programma's eindigen altijd met END.

OP-DRACH-TEN Je weet nu hoe een programma eruit ziet. In een programma kun je opdrachten zetten, deze heten: programma-opdrachten. Verder bestaan er opdrachten die de computer kent om een programma te beginnen, te laten bekijken en uit te voeren, deze noemen we directe opdrachten. Hier volgen de opdrachten die wij je zullen leren. Er zijn er nog meer maar met deze opdrachten kun je heel ver komen.

DIRECTE opdrachten	*PROGRAMMA-opdrachten.*
	PRINT
NEW	*DIM*
LIST	*INPUT*
RUN	*GOTO*
BREAK	*GOSUB en RETURN*
RESET	*REM*
	IF ... THEN
	FOR ... NEXT
	READ en DATA
	RND
	END

De programma-opdrachten komen in de volgende hoofdstukken aan bod. Dan leren we je hoe je ze kunt gebruiken in een programma. De directe opdrachten zullen we nu uitleggen. Dan weet je hoe je ervoor kunt zorgen dat de computer je programma's uitvoert.

NEW NEW is heel eenvoudig het Engelse woordje voor nieuw. Als je een nieuw programma wilt intypen, dan begin je met:

typ `NEW` (RETURN *indrukken*)

De computer maakt dan eerst zijn geheugen schoon, zodat jij je programma kunt intypen. Het lijkt een beetje op eerst je tafel of bureau opruimen en een nieuw stuk papier en een potlood pakken. Of een cassettebandje aan het begin zetten zodat je een nieuwe L.P. kunt opnemen. Na NEW is het geheugen schoon en je kunt beginnen. Typ nu het volgende kleine programmaatje eens in.

typ `NEW` (*druk op* [RETURN])
typ `1Ø PRINT "DAG ";` (*druk op* [RETURN])
typ `2Ø PRINT "ARLEEN"` (*druk op* [RETURN])
typ `3Ø GOTO 1Ø` (*druk op* [RETURN])
typ `4Ø END` (*druk op* [RETURN])

Zo maak nu het TV-scherm maar schoon. (Je weet dat je dan de [SHIFT] en [CLEAR] tegelijk in moet drukken). En whoeps, alles wat er stond is weg.
Zou je programma nu helemaal weg zijn zodat je opnieuw moet beginnen? Welnee, het programma is niet weg, je hebt het alleen maar even verstopt. Het nieuwe blaadje met je programma erop zit als het ware in de la van je bureau. Je kunt het blaadje er weer uit halen en bekijken. Je hebt daar de directe opdracht LIST voor nodig.

LIST

Als je LIST intypt verschijnt het blaadje weer op het TV-scherm en kun je het programma weer bekijken. Probeer het maar eens:

typ `LIST` (*druk op* [RETURN])

Staat er nu:

```
1Ø PRINT "DAG ";
2Ø PRINT "ARLEEN"
3Ø GOTO 1Ø
4Ø END
```

Ja, dan is het goed.
Als je heel lange programma's hebt, die niet op het TV-scherm passen, dan komen alleen de laatste regels op het TV-scherm. Geen nood want je kunt de eerste regels ook bekijken. Als je bijvoorbeeld een programma hebt

van 100 regels (ja die bestaan!) dan kun je die regels bekijken door te typen:

```
LIST 1Ø, 1ØØ
```

De computer laat je dan de regels met nummers tussen 10 en 100 zien.
Daarna kun je dan de volgende regels bekijken door te typen:

```
LIST 11Ø, 21Ø
```

Eenvoudig he?
Je weet nu wat je moet doen als je een nieuw programma wilt gaan maken (NEW typen).
Je kunt dat programma bekijken of het goed in het geheugen staat (LIST typen). Nu moet je de computer nog de directe opdracht geven om het programma uit te voeren.

RUN Dat doen we met de directe opdracht RUN.

typ RUN (*druk op* RETURN)

Zie je nu

```
DAG ARLEEN
DAG ARLEEN
DAG ARLEEN
DAG ARLEEN
DAG ARLEEN
```

over het TV-scherm gaan?
Ja? Gefeliciteerd! Je hebt nu een programma gemaakt en door de computer laten uitvoeren. RUN is het Engels voor hardlopen. En hardlopen kán de computer. In vliegende vaart zoekt de computer de eerste regel (10) op, voert die uit, gaat snel naar de volgende regel (20) net zo lang tot de computer END tegenkomt. Ook programma's van 100 regels of meer rent de computer moeiteloos langs.

BREAK Maar hoe kun je hem nu laten stoppen? Door op BREAK te drukken. Midden in het programma houdt de computer op met uitvoeren, en hij vertelt je waar hij was gebleven.

Hij zegt: BREAK IN 20 (bijvoorbeeld: het nummer waar hij was).

RESET Als je op [RESET] drukt stopt hij en maakt het scherm schoon. Je programma is niet weg. Typ maar eens LIST en druk op [RETURN] .
Je kunt met RUN weer beginnen en met [BREAK] of [RESET] weer stoppen. (Sommige computers hebben een andere stoptoets.)

Zo dit waren de directe opdrachten waarmee je programma's kunt beginnen, kunt bekijken en uitvoeren. Onthoud maar:

NEW	Het geheugen wordt dan schoon gemaakt en je kunt een nieuw programma beginnen.
LIST	De computer laat zien wat in zijn geheugen staat.
RUN	Is het startsein voor de computer om het programma te gaan uitvoeren.
BREAK	Door op [BREAK] te drukken stopt de computer midden in een programma.
RESET	Door op [RESET] te drukken stopt de computer en veegt gelijk het scherm schoon.

LEREN PROGRAMMEREN

INLEIDING

Dit tweede deel van het boek gaat over de taal BASIC. In BASIC worden programma's geschreven die de computer kan begrijpen. BASIC is een taal zoals Nederlands, Engels en Frans talen zijn. Een computer is alleen geen mens die talen als Nederlands en Engels zomaar kan leren en begrijpen. De computer moet je in zijn eigen taal vertellen wat hij moet doen, anders snapt hij het niet. Dat betekent, dat jij BASIC moet leren, want jij bent een mens en mensen kunnen wel nieuwe talen leren. Jij kunt ook – misschien doe je dat later nog – andere computertalen leren als PASCAL, ALGOL, LOGO, enz.
BASIC is de meest gebruikte taal die bijna iedere computer snapt. De kunst is dus, om als je de computer iets voor je wilt laten doen, bijvoorbeeld reken eens een som voor me uit, dat zó te doen dat hij je begrijpt en doet wat je zegt. Bedenk dan eerst in gewoon Nederlands wat je van hem wilt, en vertaal dat in BASIC.
In dit deel leer je de computertaal BASIC. Je moet dan de hoofdstukken op volgorde lezen. Als je het woordje typ ziet, moet je die regels op je eigen computer typen en doen wat er staat.
BASIC heeft maar een paar werkwoorden. Opdrachten noemen we die. Meestal staat er in elke zin (regel) één zo'n opdracht. Bijvoorbeeld het woord PRINT is de vertaling van het Nederlands: schrijf op het TV-scherm. En GOTO is de vertaling voor het Nederlands: ga naar de regel ... In ieder van de volgende hoofdstukken leer je een belangrijke opdracht. Als je deze hebt geleerd kun je de computer al veel voor je laten doen, zoals je helpen bij het huiswerk, telefoonnummers van je vrienden onthouden, uitrekenen hoe je met je geld kunt omgaan, je eigen spelletjes maken, enz.

Elk hoofdstuk ziet er hetzelfde uit. Eerst lees je welke opdracht je gaat leren, en wat je de computer met die opdracht kunt laten doen. Bijvoorbeeld:

> *Met PRINT kun je tekst op het TV-scherm krijgen*

Dan volgt een voorbeeld hoe de opdracht gebruikt wordt en de vertaling daarvan in het Nederlands. Bijvoorbeeld:

In BASIC:	*In Nederlands:*
`1Ø PRINT "DAG"`	*zet DAG op het TV-scherm*

Daarna krijg je programmaatjes met die opdracht die je moet typen. Bijvoorbeeld:

typ `1Ø PRINT "DAG"`
typ `2Ø PRINT "OSCAR"`
typ `3Ø END`

Het zal je waarschijnlijk niet altijd direct lukken de opdrachten zo te geven dat de computer je snapt. Je moet zijn taal dan nog iets beter leren spreken. Daarom staan achter elk hoofdstuk de vergissingen die je vaak maakt als je deze taal leert. Ook staat erbij hoe je de fouten kunt verbeteren. Soms zie je een TV-scherm staan.

Wat je in dit kader ziet, kun je op het TV-scherm van jouw computer verwachten.
Behalve de opdrachten tussendoor, staan er aan het eind van elk hoofdstuk opgaven die je moet kunnen maken als je de stof goed hebt gelezen. De antwoorden van deze opgaven vind je achterin het boek. *Veel succes!*

DE PRINT-OPDRACHT

PRINT

Met PRINT kun je tekst op het TV-scherm schrijven

in BASIC:	*in Nederlands:*
`10 PRINT "DAG"`	*zet DAG op het TV-scherm*
`20 X = 10`	*zet in vakje X getal 10*
`30 PRINT X`	*zet 10 op het TV-scherm*

We gaan met het PRINT-commando de computer leren om netjes 'dag' te zeggen. Het programma ziet er zo uit: (vergeet niet NEW te typen)

typ	`10 PRINT "DAG ";`	– *let op de punt/komma*
typ	`20 PRINT "OSCAR"`	– *of jouw naam*
typ	`30 GOTO 10`	
typ	`40 END`	

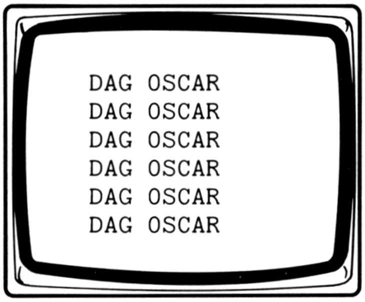
Zo, wat heb je nu gedaan?

Je hebt 4 regels in het geheugen van de computer gezet. Laten we eens kijken wat ze betekenen.

op regel 10 staat	'zet DAG op het TV-scherm'.
op regel 20 staat	'zet OSCAR op het TV-scherm'
op regel 30 staat	'ga weer naar regel 10 en begin opnieuw'
op regel 40 staat	'einde van het programma'

Als de vier regels er precies staan zoals het moet, dan volgt nu een spannend moment. Je geeft de computer de opdracht om jouw programma uit te voeren.

Typ `RUN` *(niet vergeten op* [RETURN] *te drukken)*



Als dit komt dan doet de computer precies wat jij wilde.

En de computer wil maar niet stoppen. De computer gaat net zo lang door met jouw naam op het scherm te schrijven tot dat jij zegt dat hij moet stoppen. In 'klaar om te programmeren' hebben we je uitgelegd, hoe je moet stoppen. Onderbreek het programma met BREAK of RESET.
Zo nu heb je de computer geleerd om jou gedag te zeggen. Netjes hè?

Laten we nog een voorbeeld maken.

Programma **NAAM**

```
typ  NEW
typ  1Ø PRINT  "*************"
typ  2Ø PRINT  "DIT PROGRAMMA"
typ  3Ø PRINT
typ  4Ø PRINT  "  IS VAN"
typ  5Ø PRINT
typ  6Ø PRINT  "OSCAR BAARDA"
typ  7Ø PRINT  "*************"
typ  8Ø END

typ  RUN
```

Op regel 30 en 50 staat alleen PRINT. Als je alleen PRINT typt en niets daarachter, dan krijg je er een regel tussen. Daarmee kun je wat op het scherm komt vaak mooier maken.

Je zult het al wel gezien hebben. Telkens als je tekst op het scherm wilt laten schrijven, moet je die tekst tussen aanhalingstekens " en " zetten. Dus:

```
PRINT "***********"
```

Zie je de aanhalingstekens?

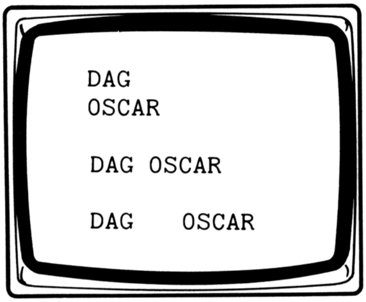
Alleen PRINT geeft een regel waar niets staat

Kijk eens naar de volgende voorbeelden

typ 1Ø PRINT "DAG" *niets erachter*
typ 2Ø PRINT "OSCAR"
typ 3Ø PRINT

typ 4Ø PRINT "DAG "; *punt/komma erachter*
typ 5Ø PRINT "OSCAR"
typ 6Ø PRINT

typ 7Ø PRINT "DAG", *komma erachter*
typ 8Ø PRINT "OSCAR"
typ 9Ø END
typ RUN



Je kunt dezelfde teksten op verschillende manieren op het TV-scherm laten komen.

—	onder elkaar	*doet hij uit zichzelf*
—	achter elkaar	*dan ; (punt/komma typen)*
—	met ruimte ertussen	*dan , (komma typen)*

> *Met PRINT kun je tekst op het TV-scherm schrijven*

herstel

1 Ben je de " (aanhalingstekens) vergeten?

Dan moet je de regel opnieuw typen.

2 Ben je de spaties vergeten?

Meestal is dat niet zo erg. Het programma draait toch wel. Het programma ziet er echter niet zo mooi uit. Het is verstandiger om de spaties toch aan te brengen. Zoals je weet, krijg je een spatie door de spatiebalk in te drukken.

3 Ben je de ; (punt/komma) vergeten?

Dan schrijft de computer de volgende regel op een nieuwe lijn en dat wou je niet. De punt/komma zorgt ervoor dat de teksten achter elkaar worden gezet. Regel opnieuw typen.

4 Heb je PINT of PRIT getypt?

Typ dan de regel opnieuw en let er op dat je PRINT typt.

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ `RUN` *(niet vergeten een* RETURN *te geven)*

opgaven

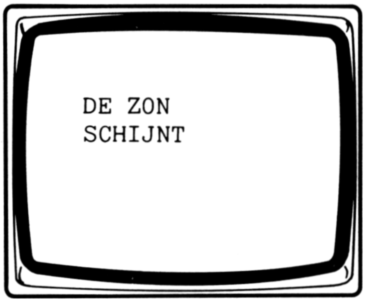
1 Zet op regel 20 de dag, bijvoorbeeld zondag, neer en kijk of je het met RUN op het TV-scherm krijgt.

typ `1Ø PRINT "VANDAAG IS HET"`
typ `2Ø ..... ...........`
typ `3Ø END`

2 Verander het volgende programma zodanig dat de teksten achter elkaar komen op het TV-scherm.

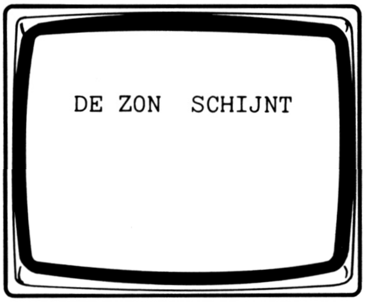
typ `1Ø PRINT "DE ZON "`
typ `2Ø PRINT "SCHIJNT"`
typ `3Ø END`

Nu staat er

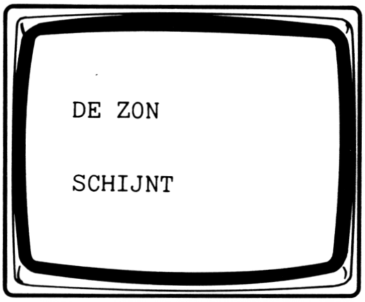


Het moet worden

3 *Nu moet het worden*



4 *Nu moet het worden*



– een regel ertussen

DE DIM-OPDRACHT

DIM

Met DIM vraag je de computer een plekje te reserveren waar letters of cijfers opgeborgen kunnen worden

In BASIC:	*In Nederlands:*
`10 DIM A(2)`	*reserveer een plekje A waar 2 cijfers kunnen staan*
`20 DIM B$(10)`	*reserveer een plekje B waar 10 letters of cijfers kunnen staan*

Laten we beginnen met een programmaatje.

```
typ 10 DIM A(2)
typ 20 A=10
typ 30 PRINT A
typ 40 END
```

Met de DIM A(2)-opdracht in dit programma vraag je de computer een plekje in zijn geheugen te maken dat hij A moet noemen en waar 2 cijfers in passen.
Dan zeg je met A=10 dat hij daar het getal 10 moet zetten, dat zijn 2 cijfers. En met regel 30 vraag je om te schrijven wat in A staat. Als je RUN typt en daarna op [RETURN] drukt zul je zien dat hij keurig het getal 10 op het scherm zet. Probeer het maar eens.
Dus met DIM reserveer je een plekje in het geheugen. Nu maakt het voor de computer verschil of hij daar cijfers of letters op moet gaan bergen. Wanneer je hem letters op wilt laten bergen, bijvoorbeeld een naam dan ziet het programma er anders uit.

```
typ 10 DIM A$(8)
typ 20 A$ = "COMPUTER"
typ 30 PRINT A$
typ 40 END
```

Zie je het verschil?

Inderdaad het is nu niet A maar A$. De $ noemen we een stringteken en daarmee zeg je dat de computer letters kan verwachten. In dit geval, DIM A$(8), moet hij plek A$ reserveren voor maximaal 8 letters.
Wanneer je letters wilt gebruiken dan moet je via DIM altijd van tevoren ruimte reserveren. Voor cijfers hoeft dat niet perse.
Let er op dat je wel voldoende plek reserveert. Kijk maar eens:

typ `10 DIM A$(3)` *nu reserveer je 3 plekjes*
typ `20 A$ = "COMPUTER"` *terwijl je er 8 nodig hebt*
typ `30 PRINT A$`
typ `40 END`

typ `RUN`

Je ziet nu dat er alleen COM komt te staan. Dat zijn de eerste 3 letters van COMPUTER waarvoor je via DIM A$(3) ruimte hebt gereserveerd.
Met DIM reserveer je plek in het geheugen van de computer. Meestal doe je dat alleen voor letters. Als er bijvoorbeeld DIM A$(8) staat reserveert hij een plek die hij A$ noemt en waar 8 letters in passen. Vergeet vooral het $ (stringteken) niet.

Met DIM vraag je de computer een plekje te reserveren waar letters of cijfers opgeborgen kunnen worden

herstel

1 ERROR 9

Als je geen DIM-opdracht hebt gegeven en je gebruikt in je programma wel een $-teken, bijvoorbeeld A$= "PIET", dan geeft hij een fout. De ATARI zal zeggen dat je ERROR 9, dat is fout nummer 9, hebt gemaakt.
Door aan het begin van je programma dan bijvoorbeeld DIM A$(25) te zetten gaat het goed. Je reserveert dan 25 plaatsen en dat is meestal voldoende.

2 Te weinig plek gereserveerd?

Kijk maar eens wat er bij dit programmaatje gebeurt.

```
1Ø DIM A$(4)
2Ø A$ = "HUISWERK"
3Ø PRINT A$
4Ø END
```

Inderdaad hij schrijft alleen HUIS, de eerste 4 letters van HUISWERK. Door bij de DIM-opdracht de 4 in een 8 te veranderen drukt hij alles af.

3 Aanhalingstekens vergeten?

Als je de aanhalingstekens vergeet kun je ook een ERROR (fout) krijgen. Bijvoorbeeld:

```
2Ø A$ = HUISWERK
```
(fout)

Zet de aanhalingstekens erbij en het gaat goed.

```
2Ø A$ = "HUISWERK"
```
(goed)

opgaven

1 Wat moet er in regel 10 staan?

```
10 ... .. ....
20 P$ = "AMSTERDAM"
30 PRINT P$
40 END
```

2 Gaat dit goed?

```
10 DIM A(3)
20 A = "AAP"
30 PRINT A
40 END
```

3 Wat is hier fout?

```
10 DIM A$(25)
20 A$ = AAP
30 PRINT A$
40 END
```

DE INPUT-OPDRACHT

INPUT

Met INPUT vraag je de gebruiker om een getal of een naam

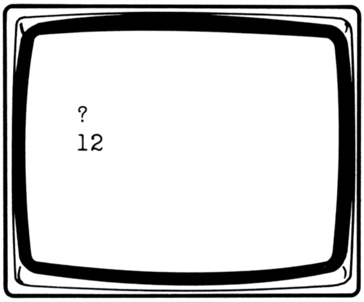
In BASIC:	*In Nederlands:*
`1Ø INPUT X`	*geef een getal*
`2Ø INPUT N$`	*geef een naam*

Met de opdracht INPUT vraag je de gebruiker vanuit het programma om een getal of een naam. Een getal kan 1, 5 of veel groter zijn. Een naam kan A, H, PIET of wat dan ook zijn. Elke combinatie van een aantal tekens is een naam. Je kunt zo complete zinnen typen.

Twee voorbeelden:

```
1Ø INPUT X
2Ø PRINT X
3Ø END
```

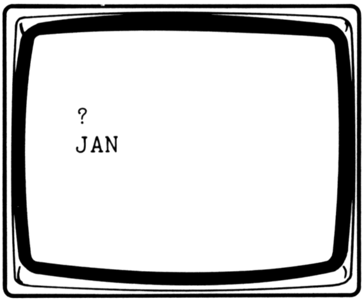
typ RUN



typ een getal

```
 5 DIM N$(15)
1Ø INPUT N$
2Ø PRINT N$
3Ø END
```

typ RUN



typ een naam

Opletten: Als je na INPUT geen $ geeft, bijvoorbeeld X, L of K, dan moet je een getal intypen en niets anders.

Als je namen wilt gaan gebruiken in je programma, moet je na de letters achter INPUT een $ zetten. De $ vindt je boven de 4.

Een $ noemen we een stringteken

Een stringteken geeft aan, dat je met andere tekens dan alleen getallen wilt werken. Als je er getallen instopt (bijvoorbeeld een telefoonnummer als 888945), kun je niet met deze getallen rekenen. De computer denkt dat het getal een naam is.

Programma **LEEFTIJD:**

```
typ  1Ø PRINT "HOE OUD BEN JIJ"
typ  2Ø INPUT L
typ  3Ø PRINT "WELK JAAR IS HET"
typ  4Ø PRINT "BIJVOORBEELD 1985"
typ  5Ø INPUT J
typ  6Ø PRINT "DAN BEN JIJ IN HET"
typ  7Ø PRINT "JAAR 2000 "; L + (2000 - J)
typ  8Ø END

typ  RUN
```

– *typ een getal, bijvoorbeeld 12*

Je ziet dat er hier gevraagd wordt om getallen via INPUT L en INPUT J. Een naam is niet mogelijk, want er staat achter L en J geen $ (stringteken).

Programma **NAAM:**

```
typ 1Ø  DIM V$(10),A$(15),J$(3)
typ 2Ø  PRINT "WAT IS JE VOORNAAM"
typ 3Ø  INPUT V$
typ 4Ø  PRINT "DANK JE WEL "; V$
typ 5Ø  PRINT "WAT IS JE ACHTERNAAM"
typ 6Ø  INPUT A$
typ 7Ø  PRINT "DUS JIJ BENT "; V$ ; " "; A$
typ 11Ø PRINT "TOT ZIENS "
typ 12Ø END

typ RUN
```

Als je zelf programma's gaat maken is het wel eens nodig dat je de computer laat vragen of je door wilt gaan. Dat kan met het volgende gedeelte dat past in het programma 'NAAM'.

```
typ 8Ø  PRINT "WIL JE STOPPEN. JA OF NEE."
typ 9Ø  INPUT J$
typ 1ØØ IF J$ = "NEE " THEN GOTO 2Ø
typ RUN
```

Met INPUT vraag je de gebruiker om een getal of een naam

herstel

1 Een punt in plaats van een komma.

`? 12,5` *(fout)*

De computer neemt alleen 12 als getal mee. Als je toch 12,5 wilt typen moet je een punt zetten in plaats van een komma, dus 12.5.

`? 12.5` *(goed)*

2 Aanhalingstekens vergeten als je $ bij INPUT hebt gebruikt. Als je in je programma bijvoorbeeld X$ gebruikt, zoals in:

```
20 IF X$ = JAN THEN GOTO 100
```
(fout)

dan moet je steeds aan de aanhalingstekens denken. Ook bij getallen.

Dus

```
20 IF X$ = 6 THEN GOTO 100
```
(fout)

```
20 IF X$ = "6" THEN GOTO 100
```
(goed)

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten* RETURN *in te drukken)*

opgaven

1 Is dit programma goed?

```
10 PRINT "WAT IS JE NAAM ?"
20 INPUT L
30 PRINT "JE NAAM IS ";L
40 END
```

2 Is dit programma goed?

```
10 DIM L$(10)
20 PRINT "WAT IS JE LEEFTIJD ?"
30 INPUT L$
40 MN = L$ * 12
50 PRINT " JE BENT ";MN; " MAANDEN JONG"
60 END
```

3 Verander programma 1 en 2 zodanig dat ze goed zijn.

DE GOTO-OPDRACHT

GOTO

Met GOTO kun je een sprong maken in het programma

in BASIC:	*in Nederlands:*
`1Ø GOTO 5Ø`	*ga naar regel 50*
`5Ø END`	*einde*

GOTO bestaat uit de Engelse woordjes GO en TO en betekent 'ga naar'. Na GOTO volgt meestal een regelnummer, bijvoorbeeld GOTO 80 of GOTO 10. Met deze opdracht kun je een sprong maken in het programma. Er zijn veel mogelijkheden:

spring naar een bepaalde regel
spring naar laatste regel
spring naar eerste regel.

Naar boven springen of naar beneden springen is allemaal mogelijk.

Voorbeeld van met GOTO springen.
Als in regel 40 X gelijk is aan R wordt er naar regel 70 gesprongen. En in regel 60 springt de computer terug naar regel 20. Je ziet dat je zowel naar boven als naar beneden kunt springen met GOTO.

```
typ  1Ø  R = 3
typ  2Ø  PRINT "RAAD EEN GETAL ?"
typ  3Ø  INPUT X
typ  4Ø  IF X = R THEN GOTO 7Ø
typ  5Ø  PRINT "PROBEER HET NOG EENS "
typ  6Ø  GOTO 2Ø
typ  7Ø  PRINT "JA, HET IS ";X
```

Typ geen NEW. Als je het volgende programma achter het vorige zet heb je een spelletje.

Voorbeeld van met GOTO naar het einde (hier regel 150) springen.

typ 5 DIM J$(3)
typ 9Ø PRINT "WIL JE STOPPEN ?"
typ 1ØØ PRINT "TYP JA OF NEE"
typ 11Ø INPUT J$
typ 12Ø IF J$ = "JA" THEN GOTO 15Ø
typ 13Ø R = R + 2
typ 14Ø GOTO 2Ø
typ 15Ø END

typ RUN

Dus je ziet dat je met GOTO naar het begin of het einde kunt springen. Maar ook naar een regel die ergens in het midden staat is mogelijk. Met GOTO kun je overal naar toe springen.

Met GOTO kun je een sprong maken in het programma

herstel

1 Naar de volgende regel springen

```
6Ø  GOTO  7Ø                              (fout)
7Ø  PRINT .......
```

Dit is niet zozeer fout, maar overbodig. Na regel 60 voert de computer zelf al de volgende regel (70) uit. Deze sprong hoef je niet te maken.

2 Naar een opdracht springen.

```
6Ø  GOTO END                              (fout)
```

Dit kan niet. De meeste computers verwachten achter GOTO een regelnummer (bijvoorbeeld 100) en niet een opdracht (bijvoorbeeld END).

```
6Ø  GOTO 1ØØ
                                          (goed)
1ØØ END
```

3 Naar zichzelf springen.

```
6Ø  GOTO 6Ø                               (fout)
```

Deze 'ga naar' springt naar zichzelf. Het gevolg is dat de computer eindeloos doorgaat. Van 60 naar 60 naar 60 naar 60 enz. Je zult hem dan met [BREAK] stil moeten zetten.

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten* [RETURN] *in te drukken)*

opgaven

1 Welke regelnummers moet je hier intypen?

```
typ 1Ø  Y = 6
typ 2Ø  PRINT "GEEF EEN GETAL TUSSEN 1 EN 1Ø"
typ 3Ø  INPUT X
typ 4Ø  IF X = Y THEN GOTO ...
typ 5Ø  PRINT "NEE, DAT IS MIJN GETAL NIET"
typ 6Ø  GOTO ...
typ 7Ø  PRINT "JA, DAT IS MIJN NUMMER "; X
```

2 Geef geen NEW na opgave 1, we gaan door met dat programma.
Welke regelnummers kun je intypen om dit programma af te maken?

```
typ 5   DIM J$(3)
typ 8Ø  PRINT "WIL JE STOPPEN ?"
typ 9Ø  PRINT "TYP JA OF NEE"
typ 1ØØ INPUT J$
typ 11Ø IF J$ = "JA" THEN GOTO ...
typ 12Ø PRINT "DUS JE WILT NIET STOPPEN"
typ 13Ø GOTO ..
typ 14Ø END
```

3 Hoe kun je het volgende programma mooier maken door van GOTO gebruik te maken?

```
typ 1Ø  N = 1
typ 2Ø  PRINT N
typ 3Ø  N = N + 1
typ 4Ø  PRINT N
typ 5Ø  N = N + 1
typ 6Ø  PRINT N
typ 7Ø  N = N + 1
typ 8Ø  PRINT N
typ 90  END
```

DE GOSUB- EN RETURN-OPDRACHTEN

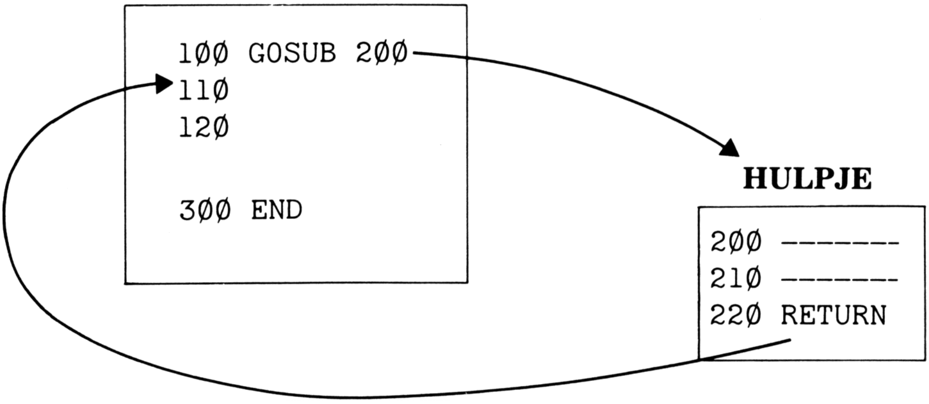
GOSUB EN RETURN

Met GOSUB maak je een sprong en met RETURN ga je weer terug

in BASIC:	*in Nederlands:*
`1Ø GOSUB 1ØØ`	*ga naar regel 100*
`2Ø`	
`3Ø`	
`1ØØ PRINT "HULPJE"`	*begin hulpprogramma*
`12Ø RETURN`	*ga terug en begin bij regel 20*

GOSUB betekent 'doe eerst het hulpprogramma vanaf regel ..'. Met GOSUB maak je een sprong naar een hulpprogramma. Ieder hulpprogramma moet eindigen met een RETURN. Dat betekent 'ga terug'. GOSUB en RETURN horen bij elkaar. Als je GOSUB gebruikt, moet je ook RETURN gebruiken.

HOOFDPROGRAMMA



Vanuit het hoofdprogramma spring je naar een gedeelte van het programma (hulpprogramma) en daarna weer terug naar de volgende regel in het hoofdprogramma.

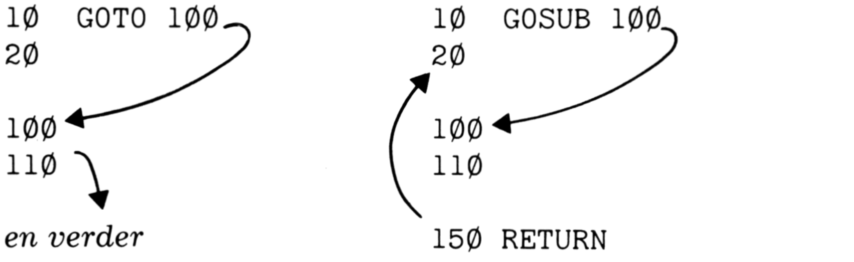
Programma **REKEN:**

```
typ  10   PRINT  "REKENPROGRAMMA"
typ  20   PRINT  "GEEF EEN GETAL"
typ  30   INPUT  G1
typ  40   PRINT  "GEEF NOG EEN GETAL"
typ  50   INPUT  G2
typ  60   GOSUB  100
typ  70   GOSUB  200
typ  80   GOTO   300
typ  100  PRINT G1;" + ";G2;" = ";G1 + G2
typ  110  RETURN
typ  200  PRINT G1;" - ";G2;" = ";G1 - G2
typ  210  RETURN
typ  300  END

typ  RUN
```

Nu zie je gelijk wat het voordeel is van GOSUB en RETURN. Je kunt er zo vaak als nodig is naar toe springen. Als je vaak hetzelfde moet doen in een programma, dan is het handig om een hulpprogramma te maken waar je iedere keer naar toe springt. GOSUB lijkt een beetje op GOTO ('ga naar'). Maar met GOSUB kom je weer terug en met GOTO niet. Met GOTO blijf je waar je bent.

Verschil GOTO en GOSUB



GOSUB en RETURN zijn handig als je vaak hetzelfde moet doen. Net zoals in sommige liedjes waar je achter ieder couplet een refrein moet zingen.

HOOFDPROGRAMMA

Alles in de wind,
alles in de wind,
't is maar een schipperskind.

GOSUB

O, wat een spijt,
o, wat een spijt,
nu is m'n zusje kwijt.

GOSUB

O, wat een nood,
o, wat een nood,
nu is mijn zusje dood.

GOSUB

Op die brug,
op die brug,
vond ik m'n zusje terug.

GOSUB

Einde

HULPPROGRAMMA

refrein:

Kom hier, Rosa,
je bent m'n zusje,
je bent m'n zusje,
Kom hier, Rosa,
je bent m'n zusje, ja, ja.

RETURN

Hier moet je op vier verschillende plaatsen het refrein 'kom hier, Rosa' doen. Dat is vier keer hetzelfde. En als je vaak hetzelfde op verschillende plaatsen moet doen kun je er een hulpprogramma van maken met GOSUB en RETURN.
Je ziet dat er vier keer GOSUB staat, en maar één keer RETURN.

Programma **GETALLEN RADEN**

```
typ 2Ø  X=7
typ 3Ø  GOSUB 1ØØ
typ 4Ø  X=4
typ 5Ø  GOSUB 1ØØ                          hoofd-
typ 6Ø  X=9                                programma
typ 7Ø  GOSUB 1ØØ
typ 8Ø  PRINT "EN NU ZIJN WE KLAAR"
typ 9Ø  GOTO 2ØØ

typ 1ØØ PRINT "WELK GETAL HEB IK ?"
typ 11Ø INPUT GETAL
typ 12Ø IF GETAL = X THEN GOTO 15Ø
typ 13Ø PRINT "PROBEER HET NOG EENS"       hulp-
typ 14Ø GOTO 11Ø                           programma
typ 15Ø PRINT "GOED ZO, DE VOLGENDE."
typ 16Ø RETURN

typ 2ØØ END

typ RUN
```

Met GOSUB maak je een sprong en met RETURN ga je weer terug

herstel

1 Een RETURN vergeten.

```
10  GOSUB 50
20  GOTO 100

50  PRINT "DIT IS REGEL 50"
```
(fout)

```
100 END
```

```
10  GOSUB 50
20  GOTO 100

50  PRINT "DIT IS REGEL 50"
60  RETURN
```
(goed)

```
100 END
```

2 GOTO in plaats van GOSUB gebruikt.

```
10  GOTO 100

100 PRINT "DIT IS REGEL 100"
```
(fout)

```
120 RETURN
```

RETURN is alleen mogelijk in combinatie met een GOSUB-opdracht. Dus:

```
10  GOSUB 100

100 PRINT "DIT IS REGEL 100"
```
(goed)

```
120 RETURN
```

3 Achter de laatste GOSUB moet altijd een regel staan. Anders springt een RETURN terug naar niets en dat kan de computer niet.

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten* [RETURN] *in te drukken)*

opgaven

1 De bedoeling is dat alle getallen X geprint worden. Waarom doet hij dat niet?

```
10  X = 7
20  GOSUB 200
30  X = 9
40  GOSUB 200
50  X = 12
60  GOSUB 200
70  GOTO 300
200 PRINT X
300 END
```

2 De bedoeling is dat alle getallen X geprint worden. Waarom doet hij dat niet?

```
10  X = 7
20  GOTO 200
30  X = 9
40  GOTO 200
50  X = 12
60  GOTO 200
70  GOTO 300
200 PRINT X
210 RETURN
300 END
```

3 Wanneer kun je beter een GOSUB en RETURN gebruiken dan een GOTO?

DE REM-OPDRACHT

REM

Via REM kun je mededelingen in het programma zetten (voor jezelf)

in BASIC:	*in Nederlands:*
`10 REM PROGRAMMA 1`	*mededeling PROGRAMMA 1*

REM is een afkorting van het Engelse woordje REMark en betekent opmerking. Wat achter REM staat wordt door het programma niet uitgevoerd. Het is meer een handigheidje. Je weet hier bijvoorbeeld dat het om programma 1 gaat.

REM doet in het programma niets. De REM-opdracht wordt overgeslagen. Je kunt er dus alles wat je wilt achter zetten, bijvoorbeeld mededelingen of een deel van het programma mee uitleggen. Vooral als je een programma aan een ander wilt geven kun je achter REM mededelingen zetten. Kijk maar eens naar het volgende programma:

```
typ 5   DIM V$(10), A$(15)
typ 10  REM DIT PROGRAMMA IS VAN AAD EN BEN
typ 20  REM VRAAG OM EEN VOORNAAM
typ 30  PRINT "WAT IS JE VOORNAAM ?"
typ 40  INPUT V$
typ 50  REM VRAAG OM EEN ACHTERNAAM
typ 60  PRINT "WAT IS JE ACHTERNAAM ?"
typ 70  INPUT A$
typ 80  REM ZET VOOR- EN ACHTERNAAM OP HET
        TV-SCHERM
typ 90  PRINT "DAG "; V$; " "; A$
typ 100 PRINT "LEUK MET JE KENNIS GEMAAKT TE
        HEBBEN"
typ 110 END

typ RUN
```

Je ziet dat de opmerking achter de REM-opdracht door de computer niet uitgevoerd wordt. Waarom is het dan nodig dat er REM-opdrachten in staan?
Met REM-opdrachten kun je het programma veel duidelijker maken. Zowel voor jezelf als voor anderen. Als je het programma later nog eens wilt bekijken dan weet je door de REM-opdrachten waar alles voor dient. Ook als je het programma aan een ander wilt geven kun je via REM-opdrachten, mededelingen in het programma zetten. Vooral voor lange programma's zijn REM-opdrachten handig. De programma's zijn dan veel duidelijker. Let er wel op dat er hoogstens 3 regels tekst achter REM komen. Meer kan niet.

Kijk maar eens naar het volgende programma. Het is precies hetzelfde als het programma dat we net gemaakt hebben.
Alleen de REM-opdrachten zijn er uitgelaten.

```
typ 5   DIM V$(1Ø), A$(15)
typ 3Ø  PRINT "WAT IS JE VOORNAAM ?"
typ 4Ø  INPUT V$
typ 6Ø  PRINT "WAT IS JE ACHTERNAAM ?"
typ 7Ø  INPUT A$
typ 9Ø  PRINT "DAG ";V$; " ";A$
typ 1ØØ PRINT "LEUK MET JE KENNIS GEMAAKT TE
        HEBBEN"
typ 11Ø END
```

Via REM kun je mededelingen in het programma zetten (voor jezelf)

herstel

1 De REM-opdracht gaat niet zo snel fout. Maximaal 3 regels achter de REM-opdracht wordt als een mededeling gezien en precies zo in de LISTing van een programma opgenomen.

opgaven

1 Maak programma LEEFTIJD op blz. 60 mooier door met REM-opdrachten er mededelingen tussen te zetten. Kies mededelingen die anderen kunnen gebruiken.

2 Doe dit ook voor programma NAAM op blz. 61

DE IF ... THEN-OPDRACHT

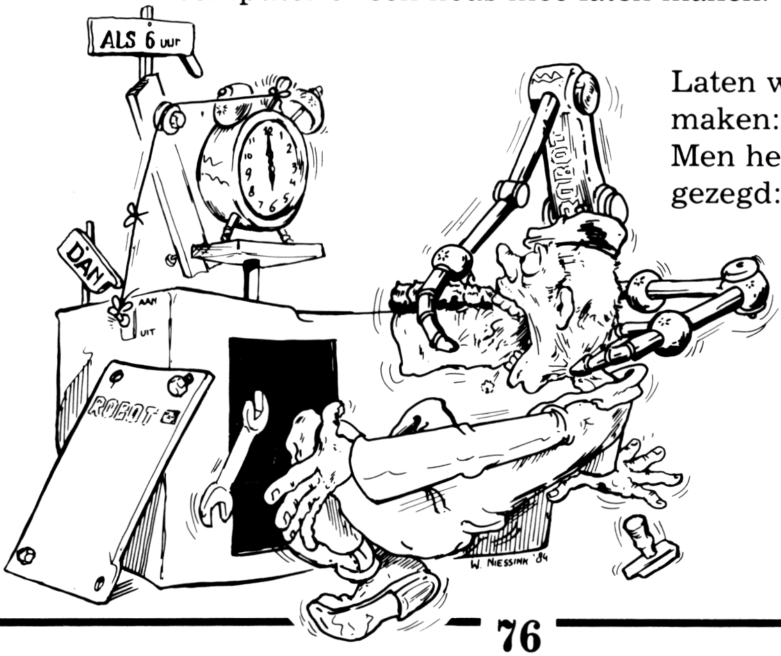
IF ... THEN

Met IF ... THEN moet de computer een keuze maken

in BASIC:	*in Nederlands:*
`1Ø   IF X = 6 THEN GOTO 1ØØ`	*als het 6 is ga dan naar regel 100*
`1ØØ END`	*einde*

IF en THEN zijn twee Engelse woordjes. IF betekent ALS en THEN betekent DAN. Zou de man die BASIC gemaakt heeft een Nederlander geweest zijn, dan had hij er vast ALS ... DAN van gemaakt. Maar die man sprak Engels en dus is het IF ... THEN geworden.

De IF ... THEN-opdracht is heel handig. Je kunt de computer er een keus mee laten maken.



Laten we een voorbeeld maken:
Men heeft tegen je gezegd: ALS het zes uur is DAN moet je gaan eten. Anders kun je doorgaan met waar je mee bezig bent.

Programma **TIJD**

```
typ  10 PRINT  "HOE LAAT IS HET ?"
typ  20 PRINT  "IN UREN"
typ  30 INPUT T
typ  40 IF T = 6 THEN PRINT  "SNEL NAAR HUIS"
typ  80 END

typ  RUN
```

– *Typ de tijd. Bijv. 5*

Alleen maar als T = 6 uur dan wordt de opdracht uitgevoerd. Als T niet 6 (uur) is dan wordt de opdracht niet uitgevoerd. Die regel wordt dan overgeslagen en gaat de computer de volgende regel uitvoeren.

Tussen IF ... THEN kunnen de volgende vergelijkingen gemaakt worden:

```
typ  40 IF T = 6 THEN PRINT  "HET IS 6 UUR"
typ  RUN

typ  50 IF T > 6 THEN PRINT  "HET IS AL 6 UUR
                                  GEWEEST"
typ  RUN

typ  60 IF T < 6 THEN PRINT  "HET IS VOOR 6 UUR"
typ  RUN

typ  70 IF T < > 6 THEN PRINT  "HET IS GEEN
                                    6 UUR"
typ  RUN
```

Dus onthouden	=	*gelijk aan*	*6 = 6*
	>	*groter dan*	*8 > 6*
	<	*kleiner dan*	*5 < 6*
	< >	*ongelijk aan*	*3 < > 6*

Er zijn nog meer vergelijkingen, maar die hoef je nog niet te onthouden.

Programma **GELUKSGETAL**

```
typ  1Ø PRINT "RAAD MIJN GELUKSGETAL"
typ  2Ø PRINT "TYP EEN GETAL TUSSEN 1 EN 1Ø"
typ  3Ø INPUT X
typ  4Ø IF X = 7 THEN GOTO 7Ø
typ  5Ø PRINT "PROBEER HET NOG EENS"
typ  6Ø GOTO 2Ø
typ  7Ø PRINT "JA, 7 IS MIJN GELUKSGETAL"
typ  8Ø END

typ  RUN
```

Als je een foutmelding krijgt dan weet je dat de computer je niet begrijpt. Eerst de vergissing herstellen en dan nog een keer RUN typen. Lukt het niet, kijk dan even in Herstel.

Programma **MIJNHEER of MEVROUW**

```
typ  5  DIM M$(1Ø), V$(1Ø)
typ  1Ø PRINT "BEN JE EEN MAN OF EEN VROUW ?"
typ  2Ø PRINT "TYP EEN M OF EEN V"
typ  3Ø INPUT M$
typ  4Ø IF M$ = "M" THEN M$ = " MENEER "
typ  5Ø IF M$ = "V" THEN M$ = " MEVROUW "
typ  6Ø PRINT "WAT IS JE VOORNAAM ?"
typ  7Ø INPUT V$
typ  8Ø PRINT "DAG "; M$; V$
typ  9Ø END

typ  RUN
```

Je ziet dat er bij INPUT weer een stringteken ($) achter de letters staat. Je hebt het nodig omdat je een naam vraagt.

> *Met IF ...THEN moet de computer een keuze maken*

herstel

1 Aanhalingstekens vergeten?

```
40 IF M$ = M THEN . . .
```
(fout)

Aanhalingstekens vergeten.

Het stringtekentje $ vraagt om een naam of letters en die moeten tussen aanhalingstekens. Herstel die regel.

typ `40 IF M$= "M" THEN PRINT M$="MIJNHEER"` *(goed)*

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ `RUN` *(niet vergeten op* RETURN *te drukken)*

opgaven

1 Verander het programma TIJD op blz. 77 zodanig dat je om 5 uur naar huis moet.

2 Wat betekent het = teken?

3 Wat betekent het > teken?

4 Wat betekent het < teken?

5 Wat betekent het < > teken?

6 Verander het programma GELUKSGETAL op blz. 78 Laat nu 3 het geluksgetal van de computer zijn.

DE FOR ...NEXT-OPDRACHT

FOR ... NEXT

Met de FOR ... NEXT-opdracht kun je de opdracht die er tussen staat herhalen

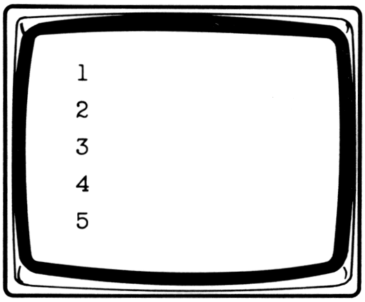
in BASIC:	*in Nederlands:*
`1Ø FOR X = 1 TO 4`	*doe van 1 tot 4 keer het volgende*
`2Ø PRINT "BEN"`	*zet BEN op TV-scherm*
`3Ø NEXT X`	*ga weer terug*

Deze opdracht wordt 4 keer herhaald.

Voorbeeld:

typ `1Ø FOR J = 1 TO 5`
typ `2Ø PRINT J`
typ `3Ø NEXT J`

typ `RUN`



Je ziet dat de FOR ... NEXT-lus een soort herhaling is van de opdracht die er tussen staat. FOR (voor) geeft het begin aan en NEXT (volgende) stuurt de computer terug naar FOR net zolang tot dat hij klaar is. 'FOR J = 1 TO 5' betekent, dat er 5 keer een opdracht moet worden uitgevoerd.

De FOR ... NEXT-opdracht wordt een lus genoemd

Je kunt een lus ook goed als een pauze in je programma gebruiken.

```
typ  10 PRINT "BEGIN PAUZE"
typ  20 FOR PAUZE = 1 TO 1000
typ  30 NEXT PAUZE
typ  40 PRINT "EINDE PAUZE"
typ  50 END

typ  RUN
```

Wat er tussen regel 20 en regel 30 staat wordt 1000 keer uitgevoerd. Maar tussen regel 20 en 30 staat geen opdracht.
De computer telt eigenlijk alleen 1000 keer en dat kost tijd. Het is net als met verstoppertje spelen. De computer moet tot 1000 tellen voordat hij mag gaan zoeken. Als de computer aan het tellen is heb jij even een pauze. Je kunt de pauze korter of langer maken door de computer korter of langer te laten tellen.

FOR PAUZE = 1 TO 100	: NEXT PAUZE	*is*	*0,2 sec.*
FOR PAUZE = 1 TO 1000	: NEXT PAUZE	*is*	*2 sec.*
FOR PAUZE = 1 TO 10000	: NEXT PAUZE	*is*	*20 sec.*

De getallen die hier genoemd worden, kunnen per computer verschillen. Vaak is het een kwestie van proberen hoe lang je de computer moet laten tellen.

Voorbeeld:

```
typ  100 FOR PAUZE = 1 TO 10
typ  110 NEXT PAUZE
typ  120 PRINT "DAG ";
typ  130 FOR PAUZE = 1 TO 100
typ  140 NEXT PAUZE
typ  150 PRINT "HOE ";
typ  160 FOR PAUZE = 1 TO 1000
typ  170 NEXT PAUZE
```

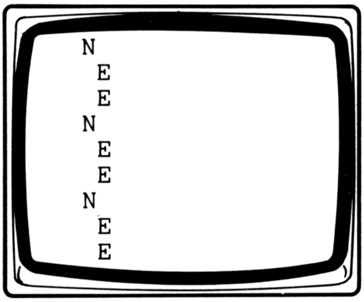
```
typ  18Ø PRINT  "GAAT HET ";
typ  19Ø FOR PAUZE = 1 TO 1ØØØØ
typ  2ØØ NEXT PAUZE
typ  21Ø PRINT  "MET JOU ?"
typ  22Ø END

typ  RUN
```

Bij deze voorbeelden staat er keurig een lus (bijv. regel 100 en 110), en daarna een andere lus (bijv. regel 130 en 140). Maar, je kunt de ene lus ook in de andere zetten. Wat je moet onthouden is, dat de computer zoekt welke FOR en NEXT het dichtst bij elkaar staan. Deze maakt hij ook het eerste af.
Kijk maar eens naar het volgende voorbeeld.

```
typ  3ØØ FOR A = 1 TO 3
typ  31Ø PRINT  "N"
typ  32Ø FOR B = 1 TO 2
typ  33Ø PRINT  " E"
typ  34Ø NEXT B
typ  35Ø NEXT A
typ  36Ø END

typ  RUN
```



Goed opletten dat je ze niet door elkaar haalt, want dan loopt je programma niet goed.

Programma **STRAFWERK**

```
typ  1Ø AANTAL = 1ØØ
typ  2Ø FOR S = 1 TO AANTAL
typ  3Ø PRINT  "IK MOET STILZITTEN"
typ  4Ø NEXT S
typ  5Ø END
```

Het voordeel van deze lus is, dat je het AANTAL makkelijk kunt veranderen. Dus als je de volgende keer 200 strafregels moet schrijven dan verander je alleen: AANTAL = 200.

Met de FOR ... NEXT-opdracht kun je de opdracht die er tussen staat herhalen

herstel

1 Heeft elke NEXT een FOR?

```
1Ø FOR J = 1 TO 5
2Ø PRINT J
```
(fout)

```
1Ø FOR J = 1 TO 5
2Ø PRINT J
3Ø NEXT J
```
(goed)

2 Staan de goede lussen bij elkaar?

```
1Ø FOR A = 1 TO 3
2Ø FOR B = 1 TO 3
3Ø NEXT A
4Ø NEXT B
```
(fout)

Lus A loopt door lus B heen. Kijk maar:

(fout)

De lussen mogen niet door elkaar lopen.

```
1Ø FOR A = 1 TO 3
2Ø FOR B = 1 TO 3
3Ø NEXT B                              (goed)
4Ø NEXT A
```

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten* RETURN *in te drukken)*

opgaven

1 Schrijf met behulp van een FOR ... NEXT-lus een programma dat de getallen 1 tot en met 10 op het TV-scherm zet.

2 Zet 10 keer jouw naam op het TV-scherm met behulp van FOR ... NEXT.

3 Verander dit programma door van een FOR ... NEXT-lus gebruik te maken.

```
typ 1Ø J = Ø
typ 2Ø J = J + 1
typ 3Ø PRINT J
typ 4Ø IF J = 1Ø THEN GOTO 6Ø
typ 5Ø GOTO 2Ø
typ 6Ø END
```

4 Met FOR ... NEXT kun je ook goed een programma schrijven dat je bijvoorbeeld laat zien wat de tafel van twee is, maar het kan ook voor de tafel van 5 of van 25 of van ...
Probeer het maar eens, achterin hebben wij er een mooi voorbeeld van.

DE READ- EN DATA-OPDRACHTEN

READ en DATA

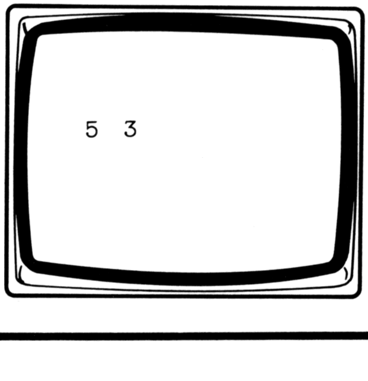
READ en DATA zijn opdrachten waarmee je zegt Lees (READ) de Gegevens (DATA) in

in BASIC:	*in Nederlands:*
1Ø READ A	*lees A (een getal)*
5Ø DATA 7	*getal is 7*
1ØØ READ A$	*lees A$ (een naam)*
15Ø DATA PIET	*naam is PIET*

READ is het Engelse woord voor lees. DATA is het Engelse woord voor gegevens. READ zoekt de gegevens op die in een DATA-opdracht staan. Net zoals bij INPUT, is een $ voor namen (of getallen waar je niet mee hoeft te rekenen). Het verschil tussen INPUT en READ is, dat INPUT gegevens vraagt aan de gebruiker. READ haalt de gegevens ergens anders uit het programma. Als er in een programma een READ-opdracht staat, gaat het programma zoeken naar een regel waar DATA staat. DATA is een ander woord voor gegevens. Met READ en DATA kun je veel leuke dingen doen.

typ `1Ø READ A, B`
typ `2Ø PRINT A, B`
typ `3Ø DATA 5, 3`
typ `4Ø END`

typ `RUN`



De READ-opdracht in regel 10 laat de computer zoeken naar een regel waar DATA staat. Het programma springt even naar regel 30. In de vakjes A en B worden dan de getallen 5 en 3 gestopt. Dan gaat het programma weer terug om de volgende regel (20) uit te voeren. Deze regel 20 zet de gegevens op het TV-scherm. Daarna is het programma klaar.
Als je net zoals hier om twee dingen vraagt (A en B), dan moeten er ook twee dingen bij DATA staan. Staat er bijvoorbeeld maar één of staan er drie, dan gaat het fout.

typ `1Ø   DATA 5, 3`
typ `3Ø   READ A, B`
typ `1ØØ  PRINT A, B`
typ `3ØØ  END`

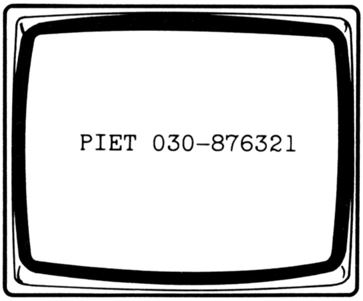
typ `RUN`

Je ziet dat het er niet toe doet, of eerst de DATA- komt en dan de READ-opdracht, of andersom. Meestal wordt wel eerst de READ-opdracht gekozen en daarna de DATA-opdracht. De DATA-opdracht staat in grote programma's vaak aan het einde.

Je kunt ook namen en telefoonnummers in de DATA-opdracht zetten. Daarmee kun je dan een telefoonlijstje maken van je vrienden.

typ `5    DIM N$ (1Ø), T$ (15)`
typ `1Ø   READ N$, T$`
typ `2Ø   PRINT N$, T$`
typ `3Ø   DATA PIET, Ø3Ø-876321`
typ `4Ø   END`

typ `RUN`



Je ziet, dat je hier het stringteken ($) moet gebruiken voor telefoonnummers.

Je begrijpt dat we met READ en DATA hele leuke programma's kunnen maken. Wat dacht je hiervan?

Programma **TELEFOONNUMMERS**

typ `5 DIM N$(1Ø), T$(15)`
typ `1Ø READ N$, T$`
typ `2Ø IF N$ = "STOP" THEN GOTO 5ØØ`
typ `3Ø PRINT N$, T$`
typ `4Ø GOTO 1Ø`

typ `1ØØ DATA PIET, Ø3Ø-123456`
typ `11Ø DATA ELS , Ø1Ø-654321`
typ `12Ø DATA HANS, Ø7Ø-112233`
typ `13Ø DATA BERT, Ø2Ø-776655`

typ `49Ø DATA STOP, STOP`
typ `5ØØ END`

typ `RUN`

Hoe werkt dit programma nu?

`5 DIM N$(1Ø), T$(15)`	*Reserveer plek onder de naam N$ en T$.*
`1Ø READ N$, T$`	*Lees de gegevens. Zoek regel waarin DATA staat.*
`2Ø IF N$ = "STOP" THEN GOTO 5ØØ`	*Als in vakje N$ "STOP" komt, ga dan naar regel 500.*
`3Ø PRINT N$, T$`	*Zet de gegevens op TV.*
`4Ø GOTO 1Ø`	*Ga weer naar regel 10 om via READ de volgende gegevens in te lezen.*

```
100 DATA PIET, 030-123456
110 DATA ELS , 010-654321
120 DATA HANS, 070-112233
130 DATA BERT, 020-776655
```

Hier staan de gegevens. Door de READ-opdracht op regel 10 worden er telkens twee gelezen.

Tussen regel 130 en 490 kun je nog veel meer namen en telefoonnummers zetten.

```
490 DATA STOP, STOP
```

Eind van de gegevens.

```
500 END
```

Eind van programma.

READ ... DATA zijn opdrachten waarmee je zegt Lees (READ) de Gegevens (DATA) in

herstel

1 Een komma te veel.

```
10 DATA 5, 10
20 READ A, B,                        (fout)
```

De komma achter B is te veel.

2 Vergeten de DATA-opdracht te geven:

```
10 PRINT 7, 10                       (fout)
20 READ A, B
```

```
10 DATA 7, 10                        (goed)
20 READ A, B
30 PRINT A, B
```

3 Te veel gegevens achter de DATA-opdracht.

```
10 DATA 5, 10, 9
20 READ A, B                         (fout)
```

Achter READ moeten net zoveel letters staan als getallen achter DATA. Als er 3 getallen staan, moeten er ook 3 letters zijn.

```
10 DATA 5, 10, 9
20 READ A, B, C                      (goed)
```

4 Komma in plaats van een punt.

```
10 READ A                            (fout)
20 DATA 25, 50
```

Als je een komma achter de 25 zet, dan wordt 50 het volgende getal. De computer denkt dat het twee getallen zijn, 25 en 50. Met READ A vertel je de computer dat hij één getal moet lezen. Dus dat gaat fout.

```
10 READ A
20 DATA 25.50                        (goed)
```

5 Naam en getal verwisseld.

```
5  DIM N$(1Ø)
1Ø READ A, N$                    (fout)
2Ø DATA PIET, 17
```

Met READ vraag je eerst om een getal voor vakje A. Maar in DATA staat eerst een naam.

```
5  DIM N$(1Ø)
1Ø READ A, N$                    (goed)
2Ø DATA 17, PIET
```

opgaven

1 Je vrienden Jan, Klaas en Wim zijn 10, 12 en 15 jaar oud. Maak een programma met READ en DATA. Laat het programma de naam en leeftijd van je vrienden op het TV-scherm zetten.

2 Je hebt een vriendin Mariska van 14 jaar erbij gekregen. Pas het programma aan.

3 Maak een telefoonlijst van je beste vrienden. Gebruik de READ- en DATA-opdracht.

4 Doe nu hetzelfde voor vrienden, familie en kennissen. (Achterin staat, hoe je dit programma op cassette kunt bewaren).

5 Wat moet er eerst in een programma gebruikt worden, READ of DATA?

DE RND-OPDRACHT

RND

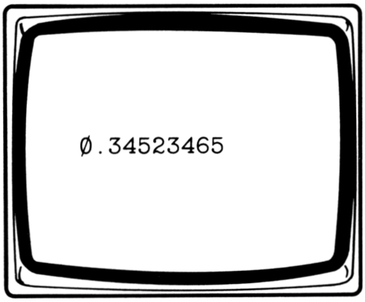
RND is een opdracht waarmee je een verrassingsgetal van de computer krijgt

in BASIC:	*in Nederlands:*
1Ø X = RND (Ø)	*zet in vakje X een verrassingsgetal*

Vooral voor veel spelletjes is het handig, als je een soort dobbelsteen hebt. Iets dat getallen geeft die je van te voren niet wist, die een verrassing zijn. Ook bij een dobbelsteen weet je van te voren niet, welk getal er boven zal liggen. RND is een soort van computerdobbelsteen, dit geeft telkens een verrassingsgetal. Dit getal ligt tussen de 0 en 1.
Typ het volgende programma maar in, dat zal het je laten zien.

typ `2Ø X = RND(Ø)`
typ `3Ø PRINT X`
typ `5Ø END`

typ `RUN`



De computer heeft hier 0.34523465 als verrassingsgetal genomen. Bij jou was het waarschijnlijk een ander getal.

En als je nog eens RUN intypt, komt er weer een ander getal.

Volgende voorbeeld:

typ 1Ø FOR I = 1 TO 5
typ 2Ø X = RND(Ø)
typ 3Ø PRINT X
typ 4Ø NEXT I

typ RUN

5 verrassingsgetallen.

Met deze getallen kunnen we nog niet zoveel. Veel mooier is het, als we getallen tussen 1 en 10 kunnen krijgen. Dat gaat op de volgende manier.

typ 1Ø X = RND(Ø)
typ 2Ø PRINT X
typ 3Ø Y = 1Ø * X
typ 4Ø PRINT Y
typ 5Ø Z = INT (Y)
typ 6Ø PRINT Z
typ 7Ø END

typ RUN

Wat hebben we nu gedaan:

Een RND getal is bijvoorbeeld		*0.34523465*
Dat nemen we 10 keer		*10 keer*
		3.4523465
Afkappen met INT	*is*	*3*

INT maakt er een heel getal van, bijvoorbeeld:

INT (3.757865) = 3
INT (2.489265) = 2

Zo, nu kunnen we de computer een getal tussen 1 en 10 laten kiezen. Het kan nog iets mooier en dat staat hieronder:

`X = INT (RND(Ø) * 1Ø    +1)` geeft getal van 1 t/m 10
`X = INT (RND(Ø) * 1ØØ   +1)` geeft getal van 1 t/m 100
`X = INT (RND(Ø) * 1ØØØ  +1)` geeft getal van 1 t/m 1000

Programma **RAAD EEN GETAL**

```
typ 1Ø PRINT "RAAD EEN GETAL VAN 1 T/M 1Ø"
typ 2Ø X = INT (RND(Ø) * 1Ø + 1)
typ 3Ø INPUT G
typ 4Ø IF G = X THEN GOTO 7Ø
typ 5Ø PRINT "NEE, NOG NIET GERADEN"
typ 6Ø GOTO 3Ø
typ 7Ø PRINT "JA, JE HEBT MIJN GETAL
            GERADEN"
typ 8Ø PRINT "HET GETAL IS "; X
typ 9Ø END

typ RUN
```

Gaat alles goed? Dan heb je een spelletje gemaakt.

RND is een opdracht waarmee je een verrassingsgetal van de computer krijgt

herstel

1 Een haakje te weinig.

Bijv. `10 X = INT ( RND(0) * 10 + 1` ← haakje vergeten

Er moeten namelijk aan de linkerkant evenveel haakjes staan als haakjes aan de rechterkant en dat klop hier niet.

opgaven

1 Maak het programma GETALLEN RADEN op blz. 70 leuker door op de regels 20, 40 en 60 een RND-getal te nemen van 1 t/m 10.

2 Hoe kun je een RND (random)-getal tussen 1 en 5 krijgen?

3 Hoe kun je een RND-getal tussen 1 en 250 krijgen?

4 Verander het programma RAAD EEN GETAL van 1 t/m 10 zo, dat hij dobbelsteengetallen geeft, dus van 1 t/m 6. (Zie voor een mooi voorbeeld blz. 93)

FOUTEN IN JE PROGRAMMA'S

Als je in je programma een fout maakt, ontdek je dat vanzelf. Typ de onderstaande regel maar eens over. Typ als je fouten in de regel ziet, ook die precies zo over. Dus je typt PINT en geen PRINT.

```
10 PINT "DIT GAAT NIET GOED"
```

Als je het hebt ingetypt en je drukt op RETURN vertelt de computer dat het fout is. Hij zegt namelijk:

```
10 ERROR- PINT ■ "DIT GAAT NIET GOED"
```

ERROR betekent namelijk fout. Je ziet ook nog een blokje in de zin staan. Dat blokje staat meestal achter het woord dat fout is. Het blokje staat hier achter PINT. Dat was fout want het moest PRINT zijn.

Nog een voorbeeld van een fout:

```
60 IF NAAM$ = PIET THEN PRINT "HOI"
60 ERROR- IF NAAM$ = PIET ■ THEN PRINT "HOI"
```

Het blokje staat achter PIET dus daar moet iets fout zijn. Zie je het? Ja PIET moet tussen aanhalingstekens " en " staan. Er staat NAAM$ dat betekent dat het om een naam gaat en die moet tussen aanhalingstekens staan.

Soms maak je ook fouten die de computer pas ontdekt als je RUN hebt getypt, dus als hij het programma uitvoert.
We zullen hier een paar van die fouten bespreken.
Typ het volgende programma eens in:

```
10 PRINT "HOE HEET JE ?"
20 INPUT NAAM
```

Als je RUN typt gaat het goed tot je een naam typt en op RETURN drukt. De computer zegt dan:

ERROR 8 ERROR 8 AT LINE 2Ø

Dat betekent dat je iets niet goed vraagt. Zie je wat fout is? Ja; NAAM. De computer verwacht een getal. Voor een naam had er NAAM$ moeten staan. Als je regel 20 verandert in

```
2Ø INPUT NAAM$
```

geeft hij nog steeds op dat er een fout is.

ERROR 9 Nu is het echter ERROR 9. Dat betekent dat je bent vergeten om plaats te reserveren in het geheugen. Dat doe je met DIM weet je nog. Je kunt de volgende regel er bij doen, dan is het goed.

```
 5 DIM NAAM$(15)
```

Je hebt nu 15 hokjes voor een naam. De naam mag dan niet langer zijn dan 15 letters.

Weet je nog wat hier mis gaat?

```
1Ø READ A, B, C
2Ø DATA 6, 7
```

ERROR 6 De computer zegt:

ERROR 6 AT LINE 10

In regel 10 vraag je de computer om 3 dingen te lezen (A, B en C). Bij DATA staan echter maar 2 getallen en het moeten er 3 zijn.

Typ nu het volgende programma eens in:

```
1Ø PRINT "HOE OUD BEN JE"
2Ø INPUT L
3Ø IF L > 3Ø THEN GOTO 8Ø
4Ø PRINT "DAN BEN JE JONG"
5Ø GOTO 7Ø
6Ø PRINT "DAN BEN JE NIET OUD"
7Ø END
```

Als je RUN intypt en je antwoordt de computer dat je bijvoorbeeld 88 bent, geeft hij als fout

ERROR 12 `ERROR 12 AT LINE 3Ø`

Er is dus iets mis in regel 30.
Dat klopt want in regel 30 moet de computer naar regel 80 gaan en die is er niet. Het had regel 60 moeten zijn.

Dit zijn de belangrijkste fouten. Je zult ze allemaal wel eens maken. Je moet maar denken, van je fouten kun je leren. En je weet nu hoe je de meest voorkomende kunt verbeteren.

DEEL 3

PROGRAMMA'S

REKENEN MET DE COMPUTER

Heb je vaak (huis)werk waarbij je moet rekenen? Als het antwoord hier ja op is, dan is het volgende iets voor jou. Je kunt de computer heel goed gebruiken bij het maken van je rekenwerk. De computer is een geduldige leraar die je goed kan helpen met leren rekenen of rekensommen maken. Optellen, aftrekken, delen of vermenigvuldigen, het maakt de computer niets uit. Grote of kleine getallen, veel sommen of weinig sommen; de computer doet het voor je. Het programma dat je daarvoor nodig hebt kun je zelf maken. Laten we eens een rekenprogramma stap voor stap opzetten. Je moet dan iedere keer als er staat typ, bijvoorbeeld:

typ `1Ø PRINT "REKENEN"`

die regel precies zo intypen. Tussen die regels zullen we stap voor stap kijken waar die regel voor nodig is en wat je verder moet doen. Aan het eind van die regels heb je een compleet rekenprogramma dat je net zo vaak kunt gebruiken als je maar wilt. Aan het eind zullen we het programma nog eens in zijn geheel geven.
Dus als je niet alle stappen wilt volgen kun je gelijk het hele programma intypen. Je kunt altijd daarna nog kijken welke stappen je gemaakt hebt en waar ze voor nodig zijn.
Zo, laten we begínnen.

Programma **REKENEN**

We willen een programma maken waarmee we kunnen optellen, aftrekken, delen en vermenigvuldigen. Allereerst typ je dan de regels waarmee je de computer vraagt of je wilt optellen, aftrekken, delen of vermenigvuldigen.

typ 1Ø PRINT "*** REKENEN *** "
typ 2Ø PRINT "TYP 1 VOOR OPTELLEN"
typ 3Ø PRINT "TYP 2 VOOR AFTREKKEN"
typ 4Ø PRINT "TYP 3 VOOR DELEN"
typ 5Ø PRINT "TYP 4 VOOR VERMENIGVULDIGEN"

Na deze regels moeten we ervoor zorgen dat de keuze in een vakje wordt opgeborgen. Laten we dat vakje de naam KEUZE geven.

typ 6Ø INPUT KEUZE

Regel 70 slaan wij even over, die gebruiken wij straks. Zo nu kunnen we in dat vakje het getal 1, 2, 3 of 4 zetten. De getallen waarmee we gaan rekenen moeten ook in vakjes komen. In het eerste vakje een getal en in het tweede vakje een getal. Die getallen kunnen we daarna bij elkaar optellen, van elkaar aftrekken, door elkaar delen of met elkaar vermenigvuldigen. Laten we de vakjes EERSTE en TWEEDE noemen.

typ 8Ø PRINT "GEEF NU TWEE GETALLEN OP"
typ 9Ø PRINT "EERSTE GETAL = "
typ 1ØØ INPUT EERSTE
typ 12Ø PRINT "TWEEDE GETAL = "
typ 13Ø INPUT TWEEDE

We hebben 3 vakjes gemaakt waarin de keuze zit (optellen, aftrekken, delen of vermenigvuldigen), het eerste getal en een vakje waarin het tweede getal zit.

Bijv.	*keuze* *1*	*eerste* *8*	*tweede* *4*

Nu moeten we de computer laten kijken of er in het vakje KEUZE een 1, 2, 3 of 4 zit.

want is het een 1 dan moet hij optellen
is het een 2 dan moet hij aftrekken
is het een 3 dan moet hij delen
is het een 4 dan moet hij vermenigvuldigen.

Zoals je weet kun je daar de IF ... THEN-opdracht voor gebruiken. IF ... THEN betekent ALS ... DAN.

Dus ALS 1 DAN +
ALS 2 DAN −
ALS 3 DAN / (delen)
ALS 4 DAN * (vermenigvuldigen)

Het getal 1, 2, 3 of 4 staat in het vakje KEUZE. En de getallen die we bijvoorbeeld willen optellen staan in de vakjes EERSTE en TWEEDE.

Als we een sommetje op papier maken ziet dat er bijvoorbeeld zo uit:

12 = 8 + 4

Getal 8 zit in het EERSTE vakje en getal 4 zit in het TWEEDE vakje. Waar moet het getal 12, de som, dan in? Daar moeten we ook een vakje voor maken en dat kunnen we SOM noemen.
We hebben nu de volgende vakjes:

Bijv.	*keuze* *1*	*som* *12*	*eerste* *8*	*tweede* *4*

We moeten nu de keuze door de computer laten maken. Als de computer het getal van de keuze (bijv. 1) gevonden heeft moet hij de optelling gaan maken. Laten we het eens uitschrijven:

als KEUZE = 1 dan SOM = EERSTE + TWEEDE
als KEUZE = 2 dan SOM = EERSTE − TWEEDE
als KEUZE = 3 dan SOM = EERSTE / TWEEDE (delen)
als KEUZE = 4 dan SOM = EERSTE * TWEEDE (verm.)

Nu nog iets aanpassen zodat de computer het begrijpt:

typ `140 IF KEUZE = 1 THEN SOM = EERSTE + TWEEDE`
typ `150 IF KEUZE = 2 THEN SOM = EERSTE - TWEEDE`
typ `160 IF KEUZE = 3 THEN SOM = EERSTE / TWEEDE`
typ `170 IF KEUZE = 4 THEN SOM = EERSTE * TWEEDE`

We zijn nu bijna klaar. De KEUZE is gemaakt. Het EERSTE en TWEEDE getal hebben we en de SOM is gemaakt. We moeten het antwoord nog op het TV-scherm krijgen. Dat gaat als volgt:

typ `300 PRINT "ANTWOORD = " ; SOM`

We zouden het programma kunnen stoppen. Als je een som moet maken of je wilt wat oefenen met rekenen dan kun je dit programma gebruiken. Iedere keer typ je RUN en het programma helpt je.
Het programma is alleen nog onhandig. Je hebt meestal meer dan één som. Soms wel 10 of nog meer. Dan is iedere keer RUN typen vervelend. We kunnen het programma heel gemakkelijk handiger maken door de computer iedere keer weer de volgende som te laten vragen. Dit gaat als volgt:

typ `310 GOTO 80`

GOTO 80 betekent ga naar regel 80. En daar staat 'geef nu de twee getallen op'. Je kunt dan gewoon met de volgende som beginnen.
Door de computer nu iedere keer opnieuw te laten beginnen heb je bereikt dat je een aantal sommen achter elkaar kunt maken. Je kunt nu maar doorgaan en doorgaan. Je moet zelfs doorgaan omdat het programma niet weet hoe te stoppen. Als je bijvoorbeeld 10 sommen hebt gemaakt en je wilt stoppen, hoe moet dat dan? Je kunt altijd de computer uitzetten of een stoptoets gebruiken, maar dat is niet mooi. Nee, laten we de mogelijkheid om te stoppen netjes in het programma opnemen.

typ `70 PRINT "JE KUNT STOPPEN DOOR TE TYPEN 999"`

Dit moeten we iedere keer controleren. Bijvoorbeeld:

als 999 dan stoppen

Het beste kunnen we dat doen na de vraag om het eerste getal (regel 90). Komt in het vakje EERSTE 999 dan wil

je stoppen. In computertaal wordt dit:

typ 11Ø IF EERSTE = 999 THEN GOTO 4ØØ

En op regel 400 moeten we dan het programma stoppen. Want GOTO 400 betekent 'ga naar regel 400'.

typ 4ØØ END

Zo, dit is het hele programma. Als je alles waar *typ* voor staat hebt ingetypt dan ben je nu klaar met het programma. Hier volgt het hele programma achter elkaar zonder commentaar erbij.

Programma **REKENEN**

```
typ 1Ø  PRINT "*** REKENEN ***"
typ 2Ø  PRINT "TYP 1 VOOR OPTELLEN"
typ 3Ø  PRINT "TYP 2 VOOR AFTREKKEN"
typ 4Ø  PRINT "TYP 3 VOOR DELEN"
typ 5Ø  PRINT "TYP 4 VOOR VERMENIGVULDIGEN"
typ 6Ø  INPUT KEUZE
typ 7Ø  PRINT "JE KUNT STOPPEN DOOR TE TYPEN
        999"
typ 8Ø  PRINT "GEEF NU TWEE GETALLEN OP"
typ 9Ø  PRINT "EERSTE GETAL =  "
typ 1ØØ INPUT EERSTE
typ 11Ø IF EERSTE = 999 THEN GOTO 4ØØ
typ 12Ø PRINT "TWEEDE GETAL =  "
typ 13Ø INPUT TWEEDE
typ 14Ø IF KEUZE = 1 THEN SOM = EERSTE + TWEEDE
typ 15Ø IF KEUZE = 2 THEN SOM = EERSTE - TWEEDE
typ 16Ø IF KEUZE = 3 THEN SOM = EERSTE / TWEEDE
typ 17Ø IF KEUZE = 4 THEN SOM = EERSTE * TWEEDE
typ 3ØØ PRINT "ANTWOORD =  " ; SOM
typ 31Ø GOTO 8Ø
typ 4ØØ END
```

Het programma is klaar en je kunt er mee rekenen. Je kunt het programma ook nog uitbreiden. Heel leuk is het om eerst zelf een antwoord te bedenken en dan de computer te laten controleren of jouw antwoord goed is.

Je gebruikt de computer dan om jou te overhoren. En je weet dat de computer heel veel geduld heeft en nooit kwaad wordt. Je kunt er net zo lang over doen als je zelf wilt en net zo vaak als je wilt.

TOETSENBORD

```
100 REM EEN KLEIN PROGRAMMA OM
110 REM VERTROUWD TE RAKEN MET
120 REM HET TOETSENBORD
130 REM
140 REM BAARDA EN VAN LONDEN
150 DIM X$(1), A$(1)
160 PRINT "NA HOEVEEL FOUTEN WIL JE STOPPEN ";
165 INPUT FOUTEN
170 IF FOUTEN = 0 THEN FOUTEN = 1
180 PRINT:PRINT "TYP HET LAATSTE TEKEN NA"
190 PRINT:PRINT "JE KUNT STOPPEN MET
    SPATIEBALK"
200 PRINT
210 FALSE = 0 : OK = 0
220 X = INT (99 * RND (0) +1)
230 IF X < 33 THEN 220
240 IF X > 90 THEN 220
250 X$=CHR$ (X)
260 PRINT "-> "; X$ ;" ";
270 OPEN #1,4,0, "K:"
280 GET #1,A : A$=CHR$(A)
290 CLOSE #1
300 PRINT A$;
310 IF A$ = X$ THEN PRINT " IS GOED " : OK=OK+1
320 IF A$ = CHR$ (32) THEN 370
330 IF A$ <> X$ THEN PRINT "IS FOUT. HET WAS
    ";X$: FALSE=FALSE+1
340 PRINT
350 IF FALSE>=FOUTEN THEN 370
360 GOTO 220
370 PRINT:PRINT "JE SCORE IS "; INT
    ((OK/(OK+FALSE)) * 100); "PROCENT "
380 END
```

DEMONSTRATIE GRAPHICS

```
10  REM DEMONSTRATIE GRAPHICS
20  A=12:B=2:C=22
30  FOR X=4 TO 8 STEP 2
40  GRAPHICS X
50  A=2*A:B=2*B:C=2*C
60  COLOR 1
70  PLOT A,B
80  DRAWTO C,A
90  DRAWTO B,A
100 DRAWTO A,B
110 PRINT "GRAPHICS ";X
120 FOR P=1 TO 1000:NEXT P
130 NEXT X
140 END
```

KLEURENDRIEHOEK

```
5   REM KLEURENDRIEHOEK
10  GRAPHICS 11
15  FOR C=1 TO 15
20  COLOR C
30  PLOT 14,32
40  DRAWTO 58,32
50  DRAWTO 36,120
60  DRAWTO 14,32
70  FOR P=1 TO 500:NEXT P
80  NEXT C
90  END
```

RANDOM TEKENING

```
3   REM RANDOM TEKENING
5   REM STOPPEN MET RESET
10  GRAPHICS 11
20  FOR Y=1 TO 300:C=RND(0)*16:COLOR C:PL
OT  40,90
30  DRAWTO INT(RND(0)*80),INT(RND(0)*192)
    :NEXT Y
40  GOTO 40
```

OLD MAC ATARI

```
10  REM OLD MAC ATARI
20  DIM A(20):X=10
30  FOR N=8 TO 1 STEP -1
40  READ B
50  A(N)=B
60  NEXT N
70  READ T,D
80  IF T=0 THEN 160
90  SOUND 1,A(T),X,X
100 SOUND 2,A(T+1),X,X
110 FOR P=1 TO D*65:NEXT P
120 SOUND 1,0,X,0
130 SOUND 2,0,X,0
140 FOR P=1 TO 25:NEXT P
150 GOTO 70
160 END
500 DATA 63,64,72,81
510 DATA 91,96,108,126
1000 DATA 5,2,5,2,5,2,5,2
1010 DATA 3,2,3,2,2,4,7,2
1020 DATA 7,2,6,2,6,2,5,6
1030 DATA 2,2,5,2,5,2,5,2
1035 DATA 5,2,3,2,3,2,2,4
1040 DATA 7,2,7,2,6,2,6,2
1050 DATA 5,6,0,0
```

DE COMPUTER IS VERLIEFD

```
10  REM DE COMPUTER IS VERLIEFD
20  REM BAARDA EN VAN LONDEN
30  DIM NAAM$(20)
40  PRINT "HALLO, HOE HEET JIJ ANDERS ? ";
50  INPUT NAAM$
60  PRINT " I ";CHR$(0); " ";NAAM$;
70  GOTO 60
80  REM ********************
90  REM * STOPPEN MET BREAK *
100 REM ********************
```

GROOTSTE GEMENE DELER

```
1Ø  REM GROOTSTE GEMENE DELER
2Ø  REM BAARDA EN VAN LONDEN
3Ø  PRINT  "GROOTSTE GEMENE DELER "
4Ø  PRINT  "EERSTE GETAL  " ;: INPUT A
5Ø  PRINT  "TWEEDE GETAL  " ;: INPUT B
6Ø  REM KIJKEN OF A KLEINER IS DAN B
7Ø  REM ZO JA, DAN A EN B OMKEREN
8Ø  IF A<B THEN H=A : A=B : B=H
9Ø  REM GGD BEREKENING
1ØØ G = INT (A/B) : R = A - (G*B)
11Ø IF R <> Ø THEN A=B: B=R : GOTO 1ØØ
12Ø PRINT:PRINT  "DE GGD IS  "; B
```

UITROLLENDE DOBBELSTEEN

```
1Ø  REM UITROLLENDE DOBBELSTEEN
2Ø  REM BAARDA EN VAN LONDEN
3Ø  REM
4Ø  X=2 : Y=135
5Ø  POKE 36878,15
6Ø  POKE 36876,Y
7Ø  Y=Y+1Ø
8Ø  FOR J=1 TO X : NEXT J
9Ø  X = X*1.2 + 5Ø
1ØØ R=INT (6*RND(Ø)+1)
11Ø PRINT R;  "  ";
12Ø POKE 36876,Ø
13Ø IF X<15ØØ THEN 6Ø
14Ø END
```

TWEE DOBBELSTENEN

```
1Ø  REM TWEE DOBBELSTENEN
2Ø  REM BAARDA EN VAN LONDEN
3Ø  PRINT  "HOEVEEL KEER WIL JE GOOIEN  ";
4Ø  INPUT N
5Ø  PRINT
6Ø  PRINT  "EERSTE TWEEDE TOTAAL"
7Ø  REM LUS VOOR AANTAL WORPEN
```

```
80  FOR I = 1 TO N
90  REM DE DOBBELSTEEN GETALLEN
100 R1 = INT (6 * RND (0) + 1)
110 R2 = INT (6 * RND (0) + 1)
120 SOM = R1 + R2
130 PRINT "  ";R1;"  ";R2;"  "; SOM
140 NEXT I
150 END
```

GETALLEN OMKEREN

```
10  REM GETALLEN OMKEREN
20  REM BAARDA EN VAN LONDEN
30  PRINT "GETALLEN OMKEREN"
40  PRINT
50  PRINT "TYP 2 GETALLEN DOOR EEN KOMMA
    GESCHEIDEN"
60  PRINT "BIJVOORBEELD 21,14"
70  INPUT A, B
80  PRINT "JOUW GETALLEN ZIJN ";
90  PRINT A;" EN  ";B
100 PRINT
110 R=A : PRINT  "RESERVE KRIJGT EERSTE GETAL =
    ";R
120 A=B : PRINT  "EERSTE KRIJGT NU DE TWEEDE =
    ";A
130 B=R : PRINT  "TWEEDE KRIJGT NU DE RESERVE =
    ";B
140 PRINT
150 PRINT  "DE COMPUTER MAAKT ERVAN  ";A;" en
    ";B
160 END
```

NAMEN OMKEREN

```
10  REM NAMEN OMKEREN
20  REM BAARDA EN VAN LONDEN
30  PRINT "NAMEN OMKEREN'
40  DIM A$(10),B$(10),R$(10)
50  PRINT "GEEF 2 NAMEN "
```

```
60  INPUT A$
70  INPUT B$
80  PRINT "JOUW NAMEN ZIJN ";
90  PRINT A$;" EN ";B$
100 PRINT
110 R$=A$ : PRINT "RESERVE KRIJGT EERSTE NAAM =
    ";R$
120 A$=B$ : PRINT "EERSTE KRIJGT NU DE TWEEDE =
    ";A$
130 B$=R$ : PRINT "TWEEDE KRIJGT NU DE RESERVE
    = ";B$
140 PRINT
150 PRINT "DE COMPUTER MAAKT ERVAN ";A$;" EN
    ";B$
160 END
```

GEDACHTENLEZEN

```
5   DIM K$(3),J$(3)
10  PRINT "GEDACHTENLEZEN "
20  PRINT "NEEM TWEE GETALLEN ONDER DE TIEN "
30  PRINT "IN GEDACHTEN "
40  PRINT "SCHRIJF ZE OP "
50  PRINT "VERMENIGVULDIG HET EERSTE CIJFER MET 5 "
60  GOSUB 300
70  PRINT "TEL HIER 7 BIJ OP "
80  GOSUB 300
90  PRINT "VERMENIGVULDIG DIT MET 2 "
100 GOSUB 300
110 PRINT "TEL HIER HET ANDERE GETAL DAT JE IN "
120 PRINT "GEDACHTEN HAD, BIJ OP "
130 PRINT "VERTEL MIJ NU WAT ER OP JE BLAADJE "
140 PRINT "STAAT ";
150 INPUT X
160 PRINT "IK MOET NU EVEN NADENKEN "
170 FOR N=1 TO 250
180 PRINT ".DENK. ";
190 NEXT N
```

```
200 G1=INT((X-14)/10)
210 G2=X-(G1*10)-14
220 PRINT
230 PRINT "VOLGENS MIJ IS HET ";G1;  "EN  ";G2
240 PRINT "KLOPT HET ? JA/NEE "
250 INPUT K$
260 IF K$= "JA " THEN GOTO 280
270 IF K$= "NEE " THEN PRINT "BEGIN DAN NOG EENS
    OPNIEUW ":GOTO 50
280 PRINT "KNAP HE "
290 END
300 PRINT "KLAAR? JA/NEE ";
310 INPUT J$
320 IF J$<> "JA " THEN GOTO 300
330 PRINT "SCHRIJF DE UITKOMST OP "
340 RETURN
```

PRIEMGETALLEN

```
10  REM PRIEMGETALLEN
20  REM BAARDA EN VAN LONDEN
30  REM
40  PRINT  "TOT WELK GETAL WIL JE "
50  PRINT  "DE PRIEMGETALLEN WETEN ";
60  INPUT MAX
70  PRINT 2,
80  FOR I = 3 TO MAX
90  Q = SQR (I)
100 N = 1
110 N = N + 1
120 IF INT (I / N) < > I / N THEN 140
130 GOTO 160
140 IF N < Q THEN 110
150 PRINT I,
160 NEXT I
170 END
```

NONSENS GENERATOR

```
5   DIM WRS$(1Ø),W$(1Ø)
1Ø  REM NONSENS GENERATOR
2Ø  REM BAARDA EN VAN LONDEN
3Ø  REM IN BASIC VOOR KINDEREN
4Ø  PRINT "DE COMPUTER IS GEK"
5Ø  PRINT "GEWORDEN. HIJ PRAAT"
6Ø  PRINT "NONSENS"
7Ø  PRINT : PRINT
8Ø  MAX = INT (5 * RND (Ø) + 1)
9Ø  IF MAX < 2 THEN 8Ø
11Ø FOR J = 1 TO MAX
12Ø X = INT (26 * RND (Ø) + 65)
13Ø W$ = CHR$ (X)
14Ø WRS$(J) = W$
15Ø NEXT J
16Ø PRINT WRS$;"  ";
2ØØ GOTO 8Ø
22Ø END
```

TELEFOONBOEK

```
5   DIM N$(2Ø),A$(2Ø),T$(2Ø)
1Ø  REM TELEFOONBOEK
2Ø  REM BAARDA EN VAN LONDEN
1ØØ PRINT: PRINT "WAT IS DE NAAM"
11Ø INPUT N$
12Ø READ A$,T$
13Ø IF A$="STOP" THEN PRINT "DIE KEN IK NIET"
    : GOTO 5ØØ
14Ø IF N$ < > A$ THEN GOTO 12Ø
15Ø PRINT "TELEFOONNUMMER= ";T$
2ØØ DATA KEES , Ø1Ø-12345
21Ø DATA MARIE, Ø1234-3344
22Ø DATA OPA , Ø7Ø-99999
3ØØ REM
31Ø REM ****************************
32Ø REM * VANAF REGEL 23Ø T/M 48Ø  *
33Ø REM * RUIMTE VOOR MEER NAMEN   *
```

```
340 REM * EN TELEFOONNUMMERS        *
350 REM ****************************
360 REM
490 DATA STOP,STOP
500 END
```

REKENKUNDIGE TAFELS

```
10  REM REKENKUNDIGE TAFELS
20  REM BAARDA EN VAN LONDEN
30  PRINT  "WELKE TAFEL WIL JE"
40  PRINT  "DE TAFEL VAN ";: INPUT X
50  PRINT
60  FOR N= 1 TO 10
70  PRINT N;" MAAL  ";X;" =  "; N * X
80  NEXT N
90  END
```

OVERHOREN TAFELS

```
10  REM OVERHOREN TAFELS
20  REM BAARDA EN VAN LONDEN
30  PRINT  "MET WELKE TAFEL WIL JE OEFENEN"
40  PRINT  "DE TAFEL VAN  ";:INPUT X
50  PRINT
60  FOR N=1 TO 10
70  PRINT N;" MAAL  ";X;" =  ";
80  INPUT A
90  IF A = N * X THEN PRINT  "JA GOED!!!"
100 IF A < >N * X THEN PRINT  "JAMMER HET IS  ";
    N * X
110 NEXT N
120 END
```

BIJLAGEN

BIJLAGE 1
DE CASSETTERECORDER

BEWAREN

Het is vervelend dat telkens als je de computer afzet je programma weg is. Je moet dan steeds opnieuw programma's intikken. Dat is ook niet nodig. Je kunt ze namelijk bewaren op een cassettebandje. Als je de volgende keer het programma wilt gebruiken, stop je de cassette in de recorder die bij de computer zit. Je geeft dan de computer de opdracht te lezen wat er op de cassette staat. En het programma zit in een paar seconden weer in het geheugen van de computer. Als je zelf een programma hebt gemaakt, kun je die ook op een cassette bewaren.

Laten wij eens een voorbeeldje nemen. Typ het volgende programma maar eens in:

```
5  DIM W$(3)
10 PRINT "IS HET MOOI WEER"
20 PRINT "TYP J OF N "
30 INPUT W$
40 IF W$ = "J" THEN PRINT "FIJN, HOPELIJK
   BLIJFT HET ZO"
50 IF W$ = "N" THEN PRINT "JAMMER HOPELIJK IS
   HET MORGEN BETER"
60 END
```

CSAVE

Wij willen dit programma bewaren. Je doet dit met CSAVE. Als je CSAVE [RETURN] typt hoor je twee pieptonen. Dat betekent dat je de REC-knop en de PLAY-knop op de cassetterecorder tegelijk in moet drukken, je moet wel stevig drukken. Eerst REC, deze vasthouden en dan PLAY indrukken. Je moet nu weer op de [RETURN] drukken.

Controleer nog even als de cassette draait of beide knoppen ingedrukt zijn, je kunt dat zo zien, en je hoort het ook, als hij geen gesuis geeft is het niet goed gegaan.

Als de computer klaar is met het programma op de cassette zetten, dan geeft hij READY en de cursor (het blokje) gaat weer knipperen. Het beeld moet er zo uit zien:

Probeer het maar eens.

LADEN

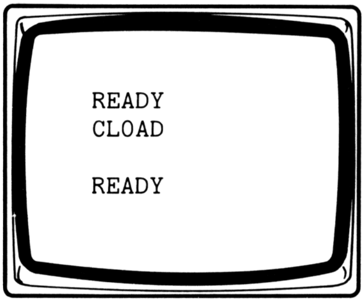
Zoals je weet, kun je je eigen programma's op een cassettebandje bewaren (zie het deel over BEWAREN). Maar je kunt ook de programma's van anderen in jouw computer laden via een cassettebandje. Het programma van een vriend bijvoorbeeld. En programma's die je in winkels koopt.

Spelletjes als PACMAN en DONKEY KONG zijn niets anders dan hele lange programma's, geschreven in een computertaal, die gebruik maken van tekenen, schrijven, rekenen en geluid maken met de computer. Stel je krijgt een cassette met PACMAN erop en je wilt dat spelletje gaan spelen. Hoe gaat dat dan?

CLOAD

Je doet dat met CLOAD. Je stopt de cassette dan in de recorder. Je moet wel de cassette terugspoelen naar waar PACMAN begint, je doet dat met de REW-knop die op de cassetterecorder zit. Als je daar op drukt spoelt hij terug. Je moet nu de computer de opdracht geven dat hij PACMAN moet laden. Je kunt dat met CLOAD doen, vergeet RETURN niet. Als het goed is hoor je nu een geluid, dat betekent dat je op de knop PLAY van de cassetterecorder moet drukken. Doe dit maar eens. Er

gebeurt nu nog niets. Hij start pas als je nog een keer op RETURN drukt. Je hoort nu een soort gesuis en dan allemaal piepjes, dat betekent dat hij leest wat er op de cassette staat. Als hij klaar is verschijnt er READY op het scherm.



Je kunt nu beginnen door RUN te typen en op RETURN te drukken. Als het goed is begint het programma dan. Klopt het niet begin dan nog eens te lezen bij CLOAD.

BIJLAGE 2

ANTWOORDEN OP DE VRAGEN

REKENEN OP DE COMPUTER

1
```
PRINT 7 + 6
13
```
(druk op RETURN *)*

```
PRINT 18 + 33
51
```
(druk op RETURN *)*

enzovoort

2
```
PRINT 7 + 6 - 3
1Ø
```
(druk op RETURN *)*

3 Let er op dat je een punt in plaats van een komma typt.

```
PRINT 2.5Ø - Ø.75
1.75
```
(druk op RETURN *)*

4
```
PRINT 5Ø - 3 * (12.5Ø - 3)
21.5Ø
```

DE PRINT-OPDRACHT

1
```
1Ø PRINT "VANDAAG IS HET"
2Ø PRINT "ZONDAG"
3Ø END
```

2
```
1Ø PRINT "DE ZON ";
2Ø PRINT "SCHIJNT"
3Ø END
```
spatie (the space before the closing quote)
punt/komma (the ; after it)

3
```
1Ø PRINT "DE ZON  ",
2Ø PRINT "SCHIJNT"
3Ø END
```
komma (the , after the quote)

4
```
1Ø PRINT "DE ZON  "
15 PRINT
2Ø PRINT "SCHIJNT"
3Ø END
```

DE DIM-OPDRACHT

1
```
DIM P$(1Ø)
```

2 Nee dat gaat niet goed.
Het stringteken voor letters staat er niet. Het moet worden:

```
1Ø DIM A$(3)
2Ø A$= "AAP"
3Ø PRINT A$
4Ø END
```

3 In regel 20 moet AAP tussen aanhalingstekens staan.

```
2Ø A$= "AAP"
```

DE INPUT-OPDRACHT

1 Dit programma is niet goed. In regel 10 wordt er een naam gevraagd. In regel 20 wordt via INPUT L een getal gevraagd. Als je een naam vraagt moet je achter L het stringteken $ gebruiken. Als er staat INPUT L$ dan verwacht de computer een naam.

2 Dit programma is niet goed.
Via regel 20 vraag je om een naam. Het $ (stringteken) betekent dat de computer alles als een naam zal behandelen. In regel 30 laat je dan de computer rekenen met een naam en dat kan hij niet.

3 Opgave 1 moet worden:

```
5  DIM L$ (25)
1Ø PRINT  "WAT IS JE NAAM ?"
2Ø INPUT L$
3Ø PRINT  "JE NAAM IS  " ; L$
4Ø END
```

Opgave 2 moet worden:

```
1Ø PRINT  "WAT IS JE LEEFTIJD ?"
2Ø INPUT L
3Ø MN = L * 12
4Ø PRINT  "JE BENT  " ; MN ;  " MAANDEN JONG"
5Ø END
```

DE GOTO-OPDRACHT

1
```
4Ø IF X = Y THEN GOTO 7Ø
6Ø GOTO 3Ø
```

Als je in regel 60 GOTO 20 getypt hebt is dat ook goed.

2
```
11Ø IF JN$ = "JA" THEN GOTO 14Ø
13Ø GOTO 2Ø
```

3 De computer moet iedere keer N = N + 1 uitvoeren. Je zou daar het volgende programma voor kunnen maken:

```
1Ø N = 1
2Ø PRINT N
3Ø N = N + 1
4Ø GOTO 2Ø
5Ø END
```

Om een programma te maken dat precies hetzelfde doet als in de opgave staat kun je ook het volgende nemen.

```
1Ø N = 1
2Ø PRINT N
25 IF N = 4 THEN GOTO 5Ø
3Ø N = N + 1
4Ø GOTO 2Ø
5Ø END
```

DE GOSUB- EN RETURN-OPDRACHTEN

1 Er staat geen RETURN achter het hulpprogramma. Als je `21Ø RETURN`
er bij zet is het programma goed.

2 De GOTO-opdrachten op regel 20, 40 en 60 moeten GOSUB-opdrachten zijn. Een RETURN zonder GOSUB werkt niet.

3 Als je vaak hetzelfde moet doen in een programma is het handiger om van dat stukje een hulpprogramma te maken. Je hoeft het dan maar één keer te schrijven. Je kunt er dan iedere keer met GOSUB naar toe springen en met RETURN weer terugkomen.

DE REM-OPDRACHT

Het gebruik van REM-opdrachten is heel persoonlijk. De een zal er veel gebruik van maken en de ander maar weinig. Je zou op de volgende manier je programma's kunnen beginnen:

```
1Ø REM DIT PROGRAMMA IS VAN ........
2Ø REM DATUM ........
```

De IF ... THEN-OPDRACHT

1 Verander regel 40 in

```
4Ø IF T = 5 THEN PRINT  "SNEL NAAR HUIS"
```

Mooier is:

```
5  Y = 5
4Ø IF T = Y THEN PRINT  "SNEL NAAR HUIS"
```

Als je nu nog eens de tijd wilt veranderen, hoef je alleen maar regel 5 te veranderen en die is een stuk korter dan regel 40.

2 =   gelijk aan

3 >   groter dan

4 <   kleiner dan

5 < > ongelijk aan

6 Verander regel 40 in

```
4Ø IF X = 3 THEN GOTO 7Ø
```

Mooier is:

```
5  Y = 3
4Ø IF X = Y THEN GOTO 7Ø
```

DE FOR ... NEXT-OPDRACHT

1

```
2Ø FOR J = 1 TO 1Ø
3Ø PRINT J
4Ø NEXT J
5Ø END
```

2

```
20 FOR J = 1 TO 10
30 PRINT "ARLEEN"
40 NEXT J
50 END
```

3

```
20 FOR J = 1 TO 10
30 PRINT J
40 NEXT J
50 END
```

4 Zie op bladzijde 114 het programma REKENKUNDIGE TAFELS.

Misschien heb je zelf zoiets gemaakt:

```
10 PRINT "WELKE TAFEL"
20 INPUT TAFEL
30 FOR J = 1 TO 10
40 PRINT J * TAFEL
50 NEXT J
60 END
```

DE READ- EN DATA-OPDRACHTEN

1

```
5   DIM N$ (10)
10  READ N$ , L
20  IF N$ = "STOP " THEN GOTO 500
30  PRINT N$ , L
40  GOTO 10
100 DATA JAN , 10
110 DATA KLAAS , 12
120 DATA WIM , 15

490 DATA STOP , 0
500 END
```

2 Zet op regel 130

```
130 DATA MARISKA , 14
```

3 Zie het programma TELEFOONNUMMERS op bladzijde 87

4 In het programma TELEFOONNUMMERS kun je van regel 140 tot en met regel 480 zoveel namen en telefoonnummers zetten als je wilt.

5 Dat maakt niet uit.
Meestal zet je eerst de READ-opdracht en daarna de DATA-opdracht.

DE RND-OPDRACHT

1 Programma GETALLEN RADEN

```
20 X = INT ( RND (0) * 10 + 1)
40 X = INT ( RND (0) * 10 + 1)
60 X = INT ( RND (0) * 10 + 1)
```

2
```
X = INT ( RND (0) * 5 + 1)
```

3
```
X = INT ( RND (0) * 250 + 1)
```

4 Je moet regel 10 en 20 veranderen.

```
10 PRINT "RAAD EEN GETAL VAN 1 T/M 6"
20 X = INT ( RND (0) * 6 + 1)
```

Heb je de smaak te pakken? Hier zijn nog drie boeken voor die fantastische ATARI...

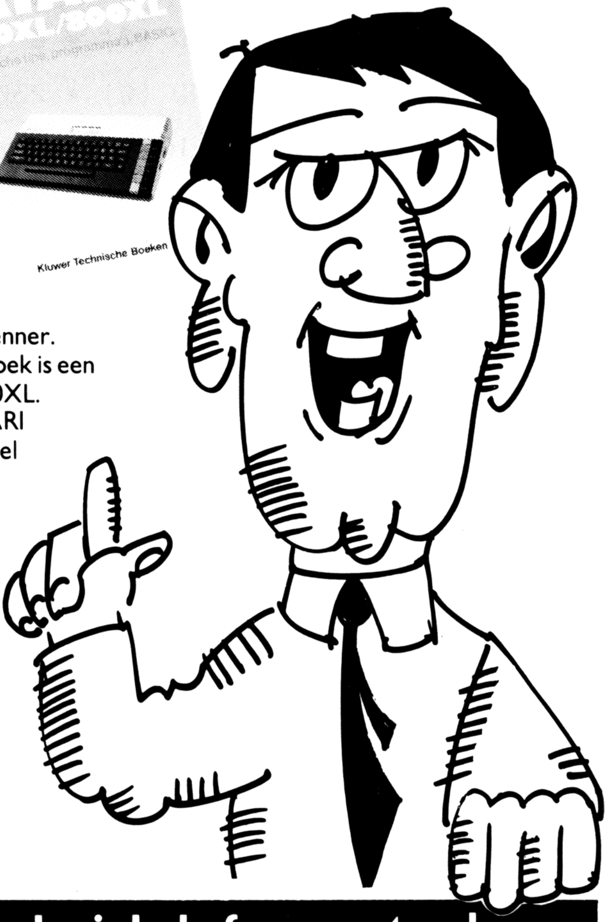
ATARI 600XL / 800XL

Praktische tips, programma's, BASIC

Albert Sickler

Albert Sickler is een heel beroemd computerkenner. Hij heeft al héél veel boeken geschreven. Dit boek is een soort handboek voor de ATARI 600XL en 800XL.
In de eerste hoofdstukken laat hij je met je ATARI kennismaken, maar dat kun je nu misschien al wel overslaan.
Dan geeft hij een beetje les in BASIC.
En dan kun je echt gaan kijken en proberen wat je ATARI allemaal kan.
Achterin staan nog een stuk of tien programma's. Bijvoorbeeld Mastermind, Dieren raden of Eeuwkalender.

Het boek heeft 192 bladzijden en kost *f* 29,50.



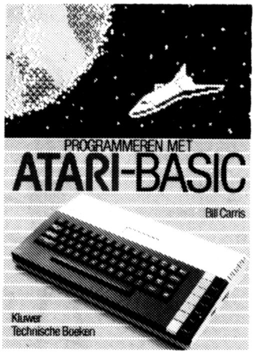
Ga eens kijken in de boekwinkel of computershop

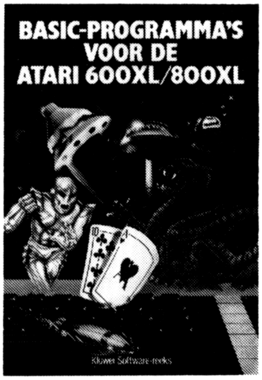
PROGRAMMEREN MET ATARI-BASIC

Bill Carris

Dit werkboek kun je gebruiken voor de ATARI 400, de 800, de 600XL, de 800XL en de 1450XLD. De schrijver van dit boek zegt: "Leren programmeren op een homecomputer is een fantastische ontdekking, een avontuur. Waarom moeilijk doen als het makkelijk kan? En fouten maken is altijd nog de beste manier om iets te leren."

Met dit boek ga je vlug, vrolijk en vriendelijk aan de gang. Als je het uit hebt ken je de BASIC, de mogelijkheden met 'graphics' en geluid en nog véél meer.
Het boek heeft 183 bladzijden en kost *f* 25,–.
Veel plezier!





BASIC-PROGRAMMA'S VOOR DE ATARI 600XL en 800XL

Peter Goode

Als je al een beetje met je ATARI overweg kunt is dit boek natuurlijk 'je van het'. Een boek vól programma's die je in kunt toetsen. En dan doet je ATARI de meest fantastische dingen. Wat dacht je van Eenentwintigen, Ruimtestad en Bioritme? Of Brandkastkraker, Onderzeebootjager, Fruitmachine of Eivanger, Wegpiraat en Schiettent?

35 verschillende programma's voor ontzéttend veel computerplezier!
Het boek heeft 128 bladzijden en kost *f* 27,50.

Vraag naar de computerboeken van Kluwer

Alle computerboeken van Kluwer staan in deze catalogus.

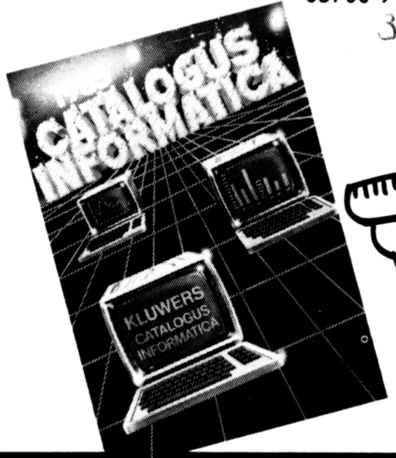
Als je echt méér wilt weten over de computer en de informatica is boeken lezen natuurlijk dé manier. Wij geven heel veel computerboeken uit. Ze staan allemaal in onze *Catalogus Informatica.* Met steeds een kleurenfoto van de voorkant erbij. Wil je ook zo'n catalogus hebben? Dat kan.

Schrijf een briefkaartje naar

Kluwer Technische Boeken BV
t.a.v. *Maja Berendsen*
Postbus 23
7400 GA *DEVENTER.*

Zij stuurt jou dan de catalogus toe, GRATIS!
Je kunt haar natuurlijk ook gewoon bellen. Haar telefoonnummer is 05700-91583.

33155



Vraag 'm aan .... hij kost niks!

Een nieuwe serie computerboeken, ditmaal speciaal voor kinderen geschreven. (Dit betekent níet dat de volwassene-er niets uit kan leren!) De auteurs, Drs. B. Baarda en Drs. A. van Londen, zijn beide werkzaam aan de subfaculteit der pedagogische en andragogische wetenschappen van de Rijksuniversiteit Utrecht.
Het boek leert een jonge computergebruiker. zo veel mogelijk zonder hulp van een volwassene, zijn computer te gebruiken.
Het eerste deel van het boek behandelt op welke wijze een computer kan tekenen, schrijven, kleuren en geluid maken.
In het tweede deel wordt bekeken voor welke toepassingen de computer nog meer is te gebruiken. Bijvoorbeeld het onthouden en bewaren van een opstel, een tekening of een muziekstukje. Ook het uitvoeren van karweitjes zoals moeilijke sommen uitrekenen, strafregels schrijven en spelletjes ontwerpen, komen aan de orde. Kortom in dit deel wordt op speelse en nuttige wijze kennis gemaakt met de programmeertaal BASIC.
Het derde deel ten slotte geeft een aantal voorbeeldprogramma's om zelf in te toetsen en te spelen. Jonge computergebruikers zullen niet alleen door de speelse benadering van de tekst, maar ook door de vormgeving zeker worden aangesproken!

ISBN 90 201 1800 5

D.B.Baarda/A.van Londen

Leren omgaan met de ATARI 600XL/800XL